Master Thesis Report

Analysis of free-cooling system for telecom data centres (Base Transceiver Stations) - big data analytics and pattern detection model

Author: Łukasz Chmielnicki

Master Program: EIT Master in Environomical Pathways for Sustainable Energy Systems

Thesis Supervisor: Roberto Villafáfila Robles

Date: June 2018

Escola Tècnica Superior d'Enginyeria Industrial de Barcelona

Table of Contents

Executive Summary

With the surging rate of digitalization across various energy sectors, the importance of data analytics and machine learning applications in broadly defined energy system is highly emerging. This thesis, taking advantage of exploratory data analysis, has led to discoveries about the distribution of data and highlighted the importance of Air Conditioning (AC) consumption and its status. Furthermore, plotting the pair dependencies chart increased the knowledge about the relations between variables, accentuated the day and night operation and displayed which values correspond to the unfavourable state of cooling system's operation. Moreover, the summary of data unravelled great discoveries in terms of energy and CO_2 emissions savings related to Free Cooling System (FCS) implementation, respectively 8928.24 kWh and 2.5 t CO_2 annually.

The most notably, with the use of machine learning algorithms, various behaviours have been exposed and grouped. One of the most prominent discoveries was "Faulty Emergency Mode", where even though FCS and AC were working with the full power, the interior temperature was either rising and surpassing the allowable limits or it was sustained flatly at a high level. This kind of pattern is undesirable as it causes increased electricity consumption and can lead to failure of servers or IT equipment, which could result in high expenses.

Moreover, one can identify the magnitude of energy, monetary and emissions savings coming from FCS implementation. Thereafter, with the use of the developed application, the operators have greater knowledge about the system behaviour and are capable of detecting anomalous activities. All of that has a potential to assure seamless and efficient operation with reduced downtime time of both FCS and AC. Additionally, the discovered situation, when AC is switched off but it still consumes occasionally some electricity can draw two conclusions, that either the AC is not working correctly and efficiently, or there are errors and short circuits in the system.

Likewise, the pattern recognition is very enriching tool as it is capable of catching very interesting insights about the sequences of behaviours that based on historical data were leading to a selected situation e.g. device's failure. This brings a higher understanding of the patterns indicating given event and shows how frequent the patterns are. As a result of this functionality, by creating a "database" of patterns the input

for further versions of the application[1] is triggered. All in all, one can easily observe that the application is bringing the deeper and more comprehensive insights and intelligence about the device's operation.

Lastly, despite the fact the program has been tested on data coming from BTS-es, the intended design of an app and the final goal was to develop a universal product to be used various industrial and commercial applications. Hence, there has been created a business case behind program's functionality. The business model assumes the deployment of the product in SaaS scheme with additional possibilities of revenues incline on the account of data science and consulting services.

[1] With real time detection and pattern recognition alerts.

Glossary

UPC - Universitat Politècnica de Catalunya

IEA – International Energy Agency

BNEF – Bloomberg New Energy Finance

MOOC - Massive Open Online Courses

KNN - K-Nearest Neighbours algorithm

CPU - Central Processing Unit

GPU - Graphics Processing unit

BTS – Base Transceiver System

FCS – Free Cooling System

ML -Machine Learning

SQL - Structured Query Language

TWh – terawatt hour

kWh - kilowatt hour

IoT – Internet of Things

EDA – Exploratory Data Analysis

AC – Air Conditioning

CAPEX – Capital Expenditures

OPEX – Operational Expenditures

CoV – Coefficient of Variance

API - Application Programming Interface

EM – Expectation -Maximization

PCA- Principal Component Analysis

HTML – Hyper Text Markup Language

GB – Gigabytes

BSD - Berkeley Software Distribution

JSON - JavaScript Object Notation

1 Introduction

The transformation of the energy industry has been accelerating rapidly for several years. The alteration from fossil fuels towards renewable sources of energy is one of the major trends that leads the change. However, there are other phenomena that can have an enormous impact on the future energy landscape – the digitalization. Energy utilities are seeking for the most optimal solutions to manage, operate and utilize their assets with the use of digital tools. It goes beyond the shadow of doubt that the energy sector has identified several fields where the implementation of software-based solutions can among others has a potential to improve efficiency and lower the cost of generation units (e.g. predictive maintenance), create new business opportunities (e.g. demand side response) or improve integration of renewables in the grid (e.g. improved forecasting) [1]. All of this possibilities are being unravelled and implemented by utilities worldwide, where the magnitude of the investments in the digitalization has been estimated by IEA to be USD 47 billion in 2016. On the other hand, BNEF's experts assessed that the benefits generated on the account of that reached USD 17 billion in 2017 and are expected to increase to the level of USD 38 billion in 2025 [2]. The various elements within the energy system in which the digitalization will play a crucial role, are presented on Figure 1 [3].

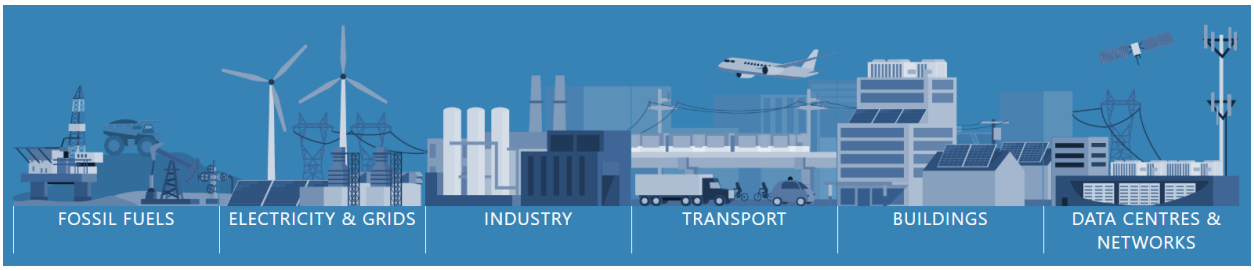

Figure 1 Energy system elements in which digitalization can play a crucial role [1].

One of the most promising applications within the digitalization concept is Big Data Analytics and Machine Learning. This thesis will focus on applying these techniques to devise a predictive maintenance model with the case presented on data centres data. There is a wide range of other inventions that are being deployed in each of the areas on Figure 1, however, the description of them goes outside the scope and will not be investigated in this paper.

1.1 Background

The master thesis has been conducted in the cooperation with Wattabit, which is a provider of an online platform which transforms energy (Watts) into information (bits), facilitating the energy management related decision-making process [4]. A company's mission is to accelerate a digital development of the energy companies, facilitate managers and any other professional committed to energy consumption optimization. All of that is being accomplished by providing the tools for a proactive management leading to an increase in efficiency, therefore reducing costs and improving the installations' profitability. A final outcome of cooperation is to build a real-life pattern detection product that is going to be implemented in the company's portfolio.

1.2 Problem identification

Regardless of the device or system considered, one would like to gain the complete knowledge about different modes of operation in order to optimally utilize its potential with the lowest possible cost. The bigger the system and the number of variables, the highest complexity one has to deal with. However, with the use of Data Analytics and Machine Learning tools, the ability of the user to acquire a greater knowledge of the device's operations can be significantly enhanced.

In the case of data centres, it is desirable to identify the situations when the devices are not working properly and to further investigate the different patterns that can lead to the abnormal or failure modes of operation. From the energy perspective, the thesis focuses on the cooling devices, which have energy needs and are responsible for controlling the temperature of servers and storage devices. Malfunction or not properly working air conditioning or fans can have a huge negative aftermath, such as loss of data or network, which results in monetary penalties for the owner of such facility. Thereby, it is essential to identify these kind of situations early enough to apply predictive measures. Hence, the thesis is focused to build an application with the use of Data Analytics and Machine Learning tool that would help with the identification of various behaviour of the devices' operation and detect the patterns that with certain probability imply anomalous performance.

1.3 Aims and objectives

The following aims have been devised in the framework of this thesis:

1. Complete the thesis within a defined deadline.
2. Acquire new knowledge about data centres structure and statistics focused on the Machine Learning field.
3. Build new skills in the fields of programming, software engineering, and web applications architecture.
4. Create a palpable real-life Machine Learning application to discover different behaviours and patterns of data centres operations.
5. Create an outlook for a further work and application development.

Having defined the aims above, Table 1 presents the objectives for each one of them.

Table 1 Project objectives description.

Aim	Objective number	Objective description
1	1.1	Create a feasible project plan and timeline management tool in the form of Gantt chart
2	2.1	Conduct a state of art research for data centres, software tools, and machine learning algorithms
3	3.1	Follow MOOCs and supplement this with online tutorials and a few books
4	4.1	Select an adequate workflow and framework for building an application
	4.2	Investigate the different modes of operation
	4.3	Investigate the data and insights coming from it
	4.4	Select an appropriate type of machine learning for the data that is available
	4.5	Make simulations for different algorithms
	4.6	Compare results, runtime, and accuracy of the algorithms
	4.7	Decide upon the models to advance within the application
5	5.1	Identify and create a list of future improvements and enhancement of the project, that could be introduced in the next versions of the application

2 State of Art

2.1 Machine Learning

Machine learning can be perceived as a particular field of computer science that with the usage of statistical methods gives the computer the capability to gradually enhance performance on a particular task. All of that is meant to be working without having a computer explicitly programmed [5]. This topic covers various techniques that in general terms are derived from the field of statistics. Thereafter, machine learning is sometimes combined with data mining methods to create an outright pipeline that enables the potential that various data can carry.

2.2 Machine Learning frameworks

There are several machine learning frameworks that are used nowadays. However, "Knowledge Discover Databases" (KDD), "Cross-Industry Standard Process for Data Mining" (CRISP-DM) and "Sample, Explore, Modify, Model, Assess" (SEMMA), presented below, are the most prominent and recognizable ones.

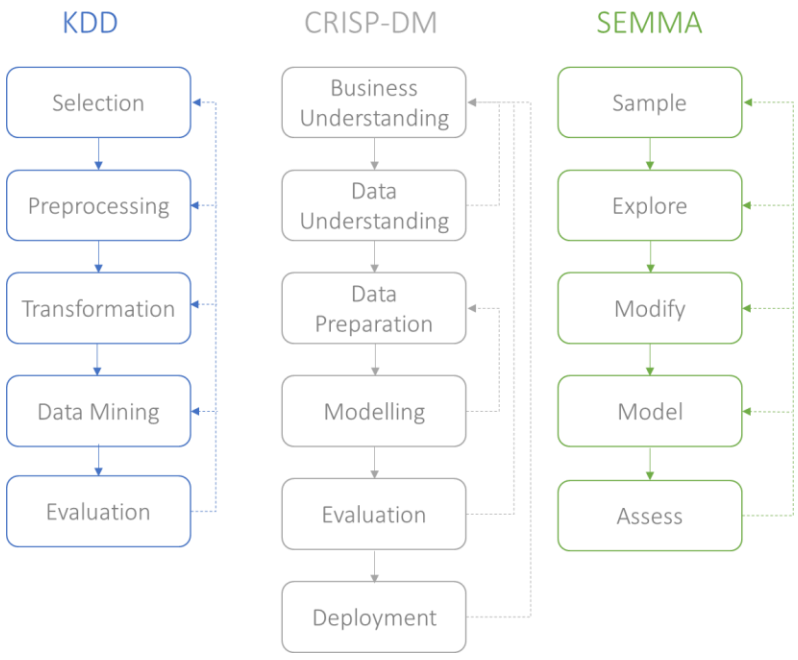

Figure 2 Machine Learning project frameworks [6].

2.2.1 KDD

This method outlines the overall process of unravelling valuable knowledge from data sets. It integrates multiple techniques for data management, machine learning, decision-making support, visualization, and computing.

A. Selection –data selection and integration from different sources.
B. Pre-processing – data cleaning process to have a data without missing values, inconsistencies and noise.
C. Transformation – data transformation an consolidation into the appropriate form for data mining process.
D. Data Mining – application of machine learning algorithms.
E. Evaluation – interpretation of data with the use of summarization and visualization to compare results for different models

2.2.2 CRISP-DM

The framework providing structure that has an unbiased methodology and is not domain dependent. It aims at consolidating data mining process with the business perspective and purpose lying behind the real-life application.

A. Business Understanding – framing a problem to be solved and gaining the overall understanding of project objectives and expectations.
B. Data Understanding – data analysis and relevance assessment of given variables.
C. Data Preparation –data cleaning for the modelling phase.
D. Modelling – different machine learning algorithms application onto the previously cleaned data.
E. Evaluation – benchmarking different algorithms with the use of various metrics, respectively for the selected type of machine learning algorithm.
F. Deployment – putting the model to the production through real-life implementation.

2.2.3 SEMMA

The method consists of a sequence of steps to create machine learning models. The technique was initially only intended to be incorporated in 'SAS Enterprise Miner', a product by SAS Institute Inc.

However, it becomes a framework that is applicable as a general development of a machine learning system.

A. Sample – a selection of the subset of data with the adequate volume and type from a larger dataset. This makes machine learning process more efficient.
B. Explore – data exploration to identify any missing values, lacks in the quality of data and relationships between variables.
C. Modify - by applying business logic to existing features new variables creation is performed.
D. Model - different machine learning algorithms application onto the data previously cleaned.
E. Assess – comparison of models' performance on the test data to assure accuracy and alignment with the business objectives.

Generally, one can observe that all three frameworks are similar to some extent. All of them intend to provide guidelines how to apply data mining techniques in real-life applications. Currently, the majority of machine learning projects in the field of research is executed accordingly to KDD and CRISP-DM due to high precision and completeness. Conversely, in the business environment, the majority takes advantage of either CRISP-DM or SEMMA to link the benefits of the more research-oriented method with the business objectives. In this thesis, a slightly modified CRISP-DM framework has been used as it encapsulates business orientation and provides the outright life cycle of creation of a machine learning application.

2.3 Introduction to Machine Learning

The main types of machine learning, including the information of the type of data that is applicable for given methods, and examples of certain algorithms are presented on the diagram below[2].

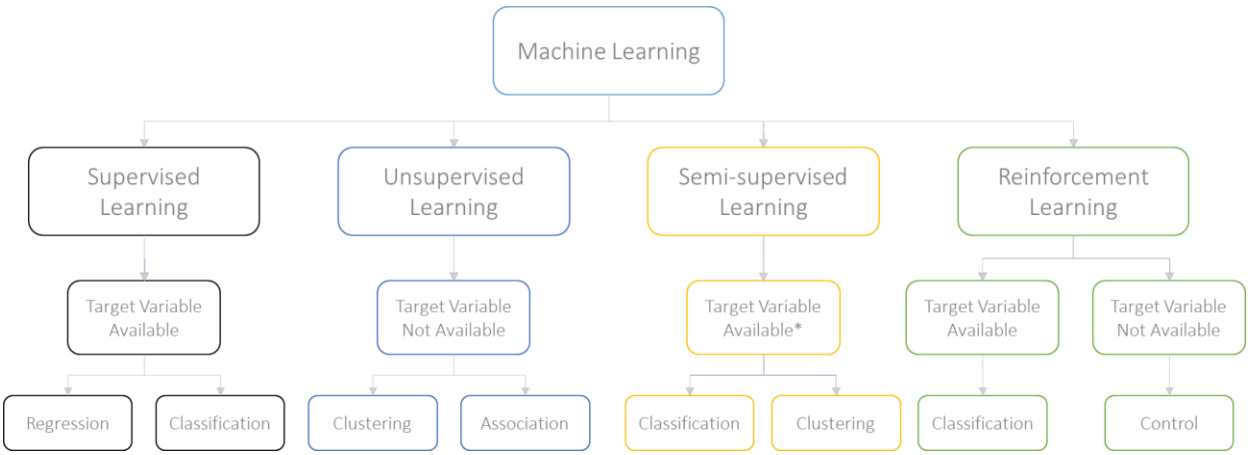

Figure 3 The diagram of machine learning types.

1. Supervised Learning – encapsulates the algorithms that endeavour to distinguish outcomes and learn by attempting to discover patterns in labelled data set. It is required to have manually labelled data, that is why a human interaction is inevitable. The technique uses a target feature (or dependent feature) which is to be foreseen from a given set of predictors (independent variables). By inputting these set of features, a function is produced to link the inputs to anticipated outputs. The training of the model lasts until the desired level of accuracy is achieved on the training set of data.

   A few examples: Linear Regression, Decision Tree, KNN or Logistic Regression.
2. Unsupervised Learning – algorithms are fed with unlabelled data (no target features) and try to detect patterns in a data without a human interaction. Assignment of the different patterns to the set of predictors is based on certain criteria, specified for each method individually.

   A few examples: K-Means, Hierarchical Agglomerative Clustering etc.
3. Semi-supervised learning- these techniques rely on a small set of labelled and big part of unlabelled data, thus it implies a necessity for human interaction. The semi-supervised estimators are capable of making a use of an unlabelled set of predictors to better identify the shape of the underlying distribution of data and generalize new samples, without being exposed to the bias of the labelled data.

[2] Techniques not covered in this topic includes inter alia ensemble learning, meta learning and deep learning, that are sometimes interchangeably account in one of the listed above types of machine learning [20].

* For semi-supervised learning the amount of target variable is very small compared to unlabeled one.

A few examples: Hidden Markov Models, Mixture of multinomial or Gaussian distributions.

4. Reinforcement learning - techniques aims at taking advantage of the observations collected from the process of interaction with the surroundings, as a result of reward maximization or minimization the risk. Algorithms are continuously learning from the environment in an iterative way, by being either rewarded for a good answer or penalized for a not correct one. The training process lasts until the agent explores the full spectrum of possible situations. In this way, machines are able to automatically distinguish the ideal behaviour within a specific context, in order to maximize its performance. A few examples: Q-Learning, Temporal Difference, or Deep Adversarial Networks etc.

2.3.1 Available Tools

The number of machine learning tools is growing very rapidly. Some of them offer a very simple interface and documentation , whereas the others provide users with the more comprehensive structure, enabling more sophisticated modelling. The major examples of the available tools are presented below:

1. Amazon Machine Learning - is a cloud-based service that enables a user with different levels of substantive knowledge to use machine learning algorithm. The tool offers a user-friendly interface to easily follow the process of creating the models and visualizing the results without having to learn complex algorithms. It provides its user with the API to do a batch or real-time predictions. It is also a complete product that allows users to build machine learning applications. However, the service is not free and works in the pay-per-use scheme.
2. Azure ML Studio - allows users to build and train models, which can be turned into APIs and afterwards consumed by other services. The free account provides user with up to 10GB of storage per account for model data. A wide spectrum of algorithms and already prepared models can be used by user or to build on top of them a new machine learning application.
3. Caffe - offers the deep learning framework with the focus on expression, speed, and modularity. Models and optimization are outlined by configuration without hard-coding. Additionally user can specify which component is used for computing - CPU or GPU. Speed is something what makes Caffe a perfect solution for doing a research experiments and quick industry deployment.
4. H2O - makes it easy for anybody to apply statistical analytics to solve most challenging problems. What is unique about H20 is its intelligent combination of special features not currently found in other machine learning platforms including: Best of Breed Open Source Technology, Easy-to-use WebUI and Familiar Interfaces, Data Agnostic Support for all Common Database and File Types. H2O is compatible with existing languages and can extend the platform into Hadoop.

5. R with different packages – R programming language offers a wide variety of packages (e.g. carnet, randomForest, e1071 etc.) to be used for machine learning processes. It provides users with strong statistical and visualization tools with the user-friendly way. Nonetheless, experience and skills are required to unleash the full potential of this software.
6. MLlib (Apache Spark) - is a machine learning library. It scales nicely and allows user to make a practical use of machine learning algorithms in an easy way. It includes most common machine learning techniques together with lower-level optimization primitives and higher-level pipeline APIs.
7. Mlpack - is a machine learning library based on C++. It has been designed to provide scalability, speed, and robustness. Use of this library can be implemented through a cache of command-line for trail or fast check of functionality. Moreover, it can operate as a "black box", or with a C++ API for more advanced applications.
8. Scikit-Learn - is a machine learning library based on Python's existing NumPy, SciPy, and matplotlib libraries. It is available under a BSD license, thus is available for all users. Scikit-learn consists of various machine learning methods. Additionally, since it is developed by a large community of experts, new updates and techniques tend to be implemented in a fairly short period of time.
9. Shogun was created in 1999 and is one of the oldest machine learning libraries. Despite the fact that is has been created in C++, it is not limited to working only with this language. The SWIG library makes it open to be used in Java, Python, C#, Ruby, R, Lua, Octave, and Matlab. Shogun is designed for more general applications that require only basic algorithms.
10. TensorFlow is an open source library that uses data flow graphs. It is implemented, where portions of data (so called "tensors") can be handled by a series of algorithms explained by a graph. The transfers of the data through the system are called "flows". TensorFlow includes machine and deep learning algorithms, can be can be used with C++ or Python can be computed with the use of either CPUs or GPUs.
11. Theano is an open-source and numerical computation library for Python. It utilizes the functionality of NumPy's mathematical expressions. Theano is intended to be used for problems involving large amounts of data where it can run efficiently on either CPU or GPU architectures.

2.3.2 Available data storage methods

Database is nothing more but the organized collection of data, where user can access, review, and update particular pieces of information in a rapid and coherent manner. In general, taking into consideration which type the data is being stored, there can be distinguished two types of databases: relational (so-called "SQL") and non-relational (so-called "NoSQL") databases. The graphical comparison is shown on Figure 4.

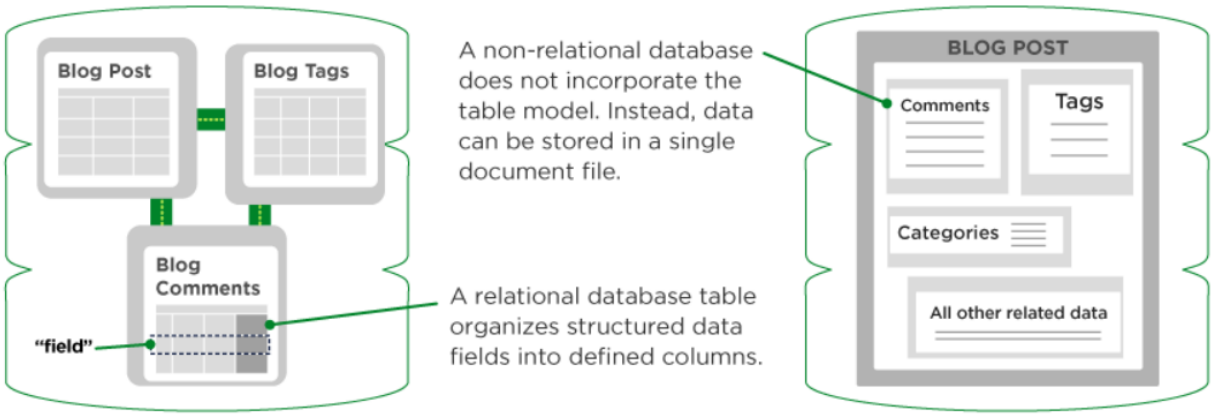

Figure 4 SQL and NoSQL databases – graphical comparison [7].

2.3.2.1 Relational Database Management Systems (RDBMS)

Relational databases store data according to scheme that allows displaying the data in the form of tables with rows and columns. RDBMS has fixed attributes and types of data that can be stored in one table and it provides a functionality for reading, creating, updating, and deleting data, typically with the use of Structured Query Language (SQL) statements. The most important aspect of the tables is their key which is used to link specific columns or rows of one table with another. Furthermore, it facilitates instant access to a selected table, row, or column of interest. RDBMS is very mature technology, widely spread and used everywhere. However, it encounters difficulties in working with unstructured data and mapping one-to-one with an object or class that represents the same data.

The number of different relational databases has been growing since 1970s, however the ones listed below have become the most commonly used [8]:

- Oracle Database
- MySQL
- Microsoft MySQL Server
- PostgreSQL
- DB2

2.3.2.2 Non-relational Database Management Systems

Non-relational databases have started to be an interesting alternative to relational databases because of their growth in complexity of web applications. NoSQL databases is very flexible when it comes to the form of data storage. Thus, the critical difference between allowing unstructured data to be stored and modified. The following types of NoSQL databases can be distinguished:

- Key-Value Stores (e.g. Redis and Amazon DynamoDB) – robust database management systems. It stores solely pairs with associated key-value and provides basic range of capabilities for retrieving the value related with a known key.
- Wide Column Stores (e.g. Cassandra and Scylla) – database management system with schema-agnostic schemes that allow to stock data in families of columns or tables, where a single row can be perceived as a record or a multi-dimensional key-value.
- Document Stores (e.g. MongoDB and Couchbase) – database that do not have schema systems and stores data in the form of JSON documents. These records can be perceived as similar form to key-value or wide column stores, but the key in this case is the document name and the content can be seen as the values, regardless of the type of objects it coins.
- Graph Databases (e.g. Neo4J and Datastax Enterprise Graph) – database which architecture can be compared to a network of connected nodes or objects , with the aim to simplify data visualizations and graph analytics. In a graph database, a node or object contains data (regardless of the form) that is linked by relationships and aggregated according to markers. Graph-Oriented Database Management Systems has been created to bring out the connections between data points.
- Search Engines (e.g. Elasticsearch or Splunk) stocks data in form of-free JSON documents. The concept is similar to document stores, but with a greater importance on accessibility of unstructured or semi-structured data easily with the use of text-based searches.

2.4 Telecom-based Data Centres

This subchapter covers simplified state of art for data centres with the particular focus to present the design, structure and operation of cooling system. The description and schemes of processing, storing and transmitting digital information devices will not be enclosed in this work.

The electricity consumption used for cooling IT equipment can reach up to 50% of the total consumption of the telecom base stations (BTS) [9]. Hence, the cooling costs are one of the greatest contributors to the overall electricity bill of data centres. The devices that are responsible for keeping the temperature below

certain threshold can be perceived in general terms as air-conditioning systems (AC). In spite of the fact that these devices fulfil their role outrightly, they consume substantial amount of electricity. Thereafter, there is a need to provide a solution that would decrease the electricity requirement for cooling purposes. One of these solutions is Free-Cooling system that allows to reduce the electricity demand.

2.4.1 BTS Telecommunication Equipment

BTS's role is to facilitate the wireless communication between user device (e.g. cell phone) and a network. A regular BTS is supposed to have the following elements [10]:

- Transceiver
- Combiner
- Antenna
- Control Function
- Clock Module
- Power Amplifier
- Multiplexer
- Baseband Receiver Unit
- Alarm extension system
- Operation and Maintenance module
- Cooling System

2.4.2 Scheme

The BTS consists of several devices which role is to process, store and transmit the information from the telecommunication network that each of us is using every day. For the sake of simplicity, on Figure 5 this equipment is presented as a "black box". In general, the FCS unit consists of at least two fans (one outlet and one inlet), a gravity damper and a filter. On the example showed below, the free-cooling system has two inlets with fans mounted in the walls. The lower one is an inlet with the cool air, whereas the upper one is an outlet for a warm air. AC system is installed inside the station[3].

[3] In this example two units placed on the walls.

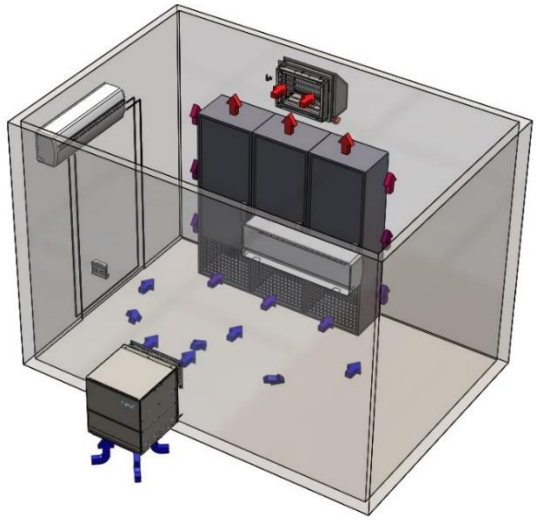

Figure 5 Simplified scheme of BTS interior with cooling systems [11].

2.4.3 Free-cooling system

In principle, free cooling system (FCS) utilizes ambient air temperature to lower the temperature inside the BTS and thus is one of the most efficient solutions to lower down the cooling power consumption in most of the regions. The operating principle of the system is based on the fact that if atmospheric temperature is under certain level, the free cooling system replaces the AC to cool down the room. Figure 6 presents an operational principle on the inside temperature versus time chart for a BTS with two ACs (CDZ acronym on the chart) and FCS (FC acronym on the chart) units. The operation regions, where temperature is below certain threshold and FCS is working, with no need for AC to cool the room. The other region appears when AC is working on and FCS is switched off.

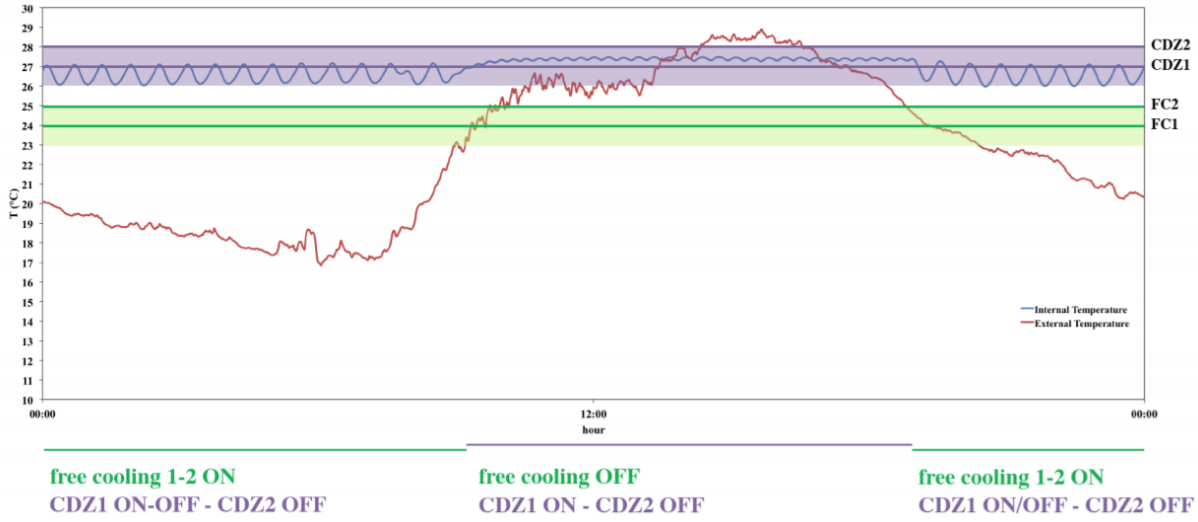

Figure 6 Operational principle example of FCS [12].

The use of FCS is enormously beneficial as the energy efficiency ratio could be 10 times higher than the AC. As a result of that the annual electricity savings could reach from 30% to 80%, depending on the region's climate situation [9]. Since, the share of electricity consumption for cooling purposes can reach the level of 50% and the total hourly electricity consumption is in the magnitude of several to tens of kWh, depending on the size of BTS. Thereby, from the economic point of view, generating great savings in terms of CAPEX and OPEX is plausible.

3 Methodology

This thesis concentrates on applying exploratory data analysis and machine learning methods in order to build a physical application which is capable of detecting various types of behaviour that might be occurring in data centre. Research adopts these techniques with the use of the data gathered from data centre's cooling system and including electricity consumption. Along the process of ideation, the following methodology has been applied to assure the proper execution of all phase of the project.

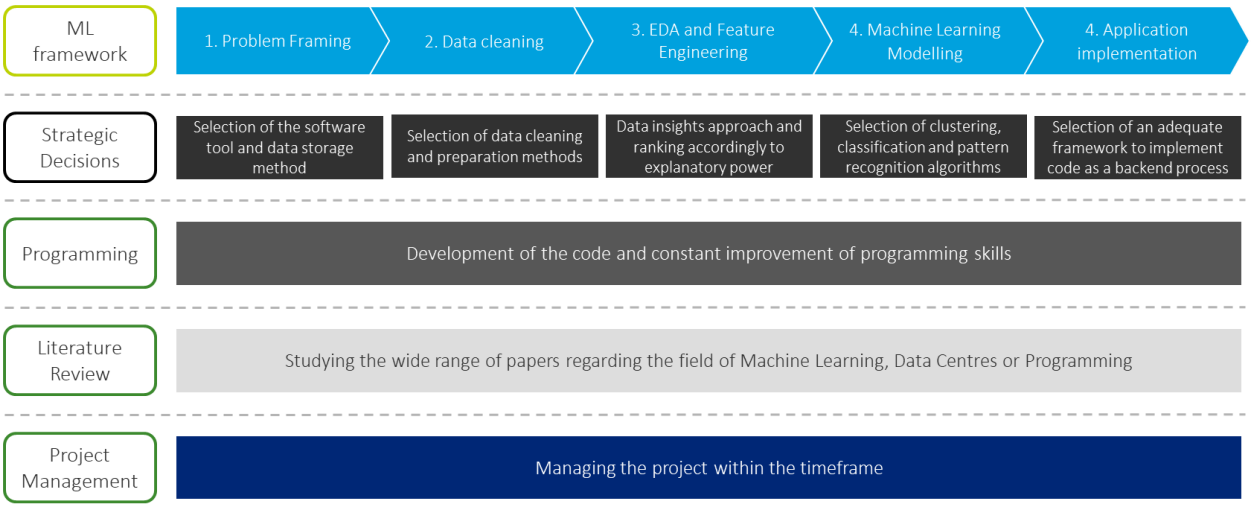

Figure 7 The general schema of master thesis methodology.

The methodology consists of 5 blocks:

- Project framework, which displays five major substantive stages of the thesis,
- Strategic decisions, that have had to be make in each of the stages,
- Three blocks that were present through the whole process of creation and development of the thesis:
    - Programming, which illustrates the development of the code to run all processes and a necessity to constantly optimize code and personal programming skills,
    - Literature review, which was crucial to support design and decision-making process,
    - Project Management, which was inevitable to optimally distribute the tasks and time.

3.1 Proposed Machine Learning Flow

The purpose if this chapter is to describe the methodology of the project. The applied methodology has been designed on the foundation of CRISP-DM framework, explained in the Subchapter 2.2.2. The major focus is to show the logical flow of the work and processes and to justify selected pathway. It consists of three main points:

- Problem framing description together with tools selection,
- Machine Learning framework that covers Data Cleaning, EDA and Feature Engineering and Machine Learning Modelling stages,
- Application Implementation that shows the final product incorporated in the company's application.

3.2 Problem framing

Together with supervisors from Wattabit and UPC, the problem was framed and the project plan was drawn with the regard to the requirements of the master thesis. Since the problem circulates around discovering of new type of behaviour, data is not labelled. Thereby, the problem has to be addressed firstly with the use of unsupervised machine learning algorithms (clustering). Later on, after having data labelled by the industry expert, supervised learning algorithms (classification) can be introduced to classify the new data.

The complete description of this stage can be found in the Subchapter 1.2.

3.3 Tools Selection

In this stage, the necessary tools – programming language, libraries and database, were selected. The decisions have been made in regard with the following criteria:

1. Capability to write a program that would deliver desired results.
2. Comprehension of available methods and algorithms within programming language, library or database.
3. Availability of wide range of libraries for programming language.
4. Completeness of the tool to design and develop all stages of the project efficiently.
5. Compatibility with the company's software environment.
6. Facility and robustness to minimise time of operation.
7. Speed of computations and operations.

Having considered all of the factors listed above, the following selection, listed in order: programming language, libraries, and database was made:[4]

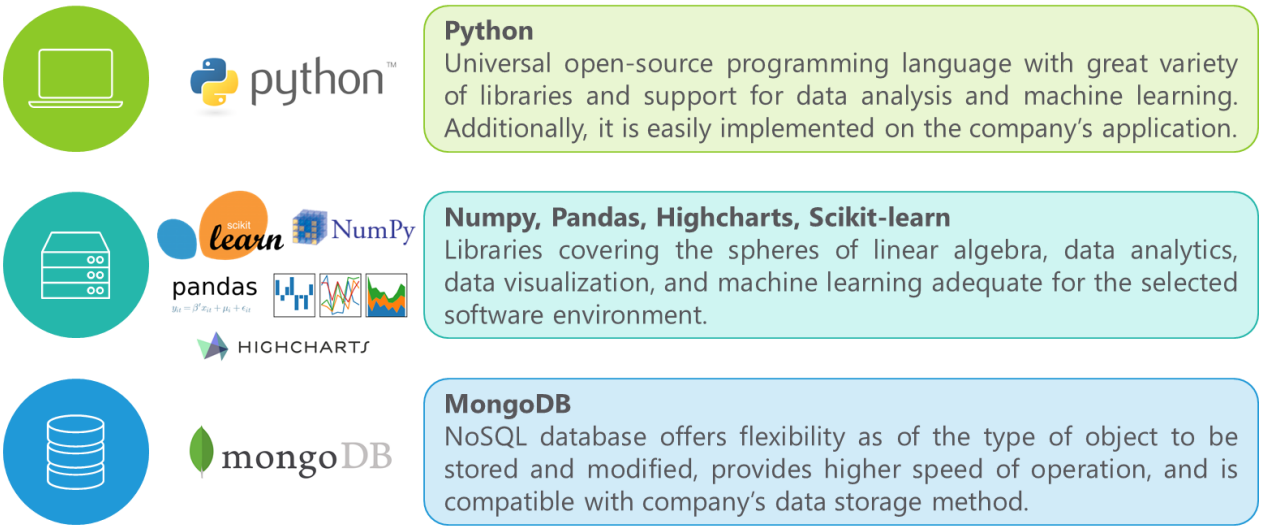

Figure 8 The tools selected in the master thesis.

3.4 Machine Learning Project Framework

After the problem, has been clearly identified and the tools have been carefully selected, the next stage corresponds to the development of program. In order to successfully accomplish that, several steps, were needed to be taken. These steps can be observed on Figure 9 hereunder.

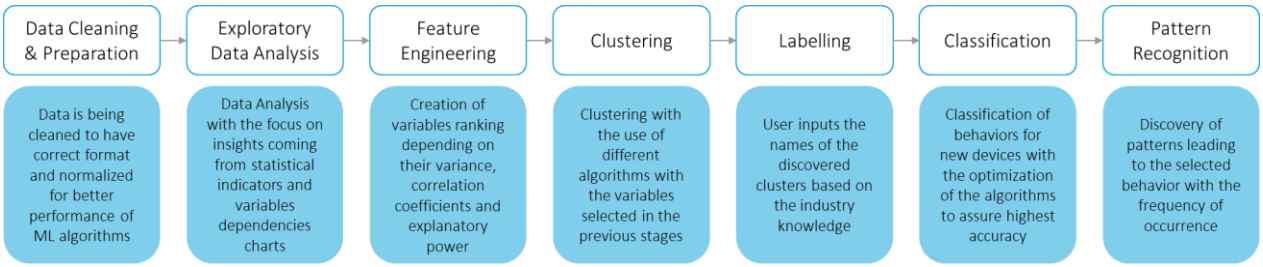

Figure 9 The machine learning project framework applied in the master thesis.

3.4.1 Data cleaning and preparation

Firstly, a connection with MongoDB database was essential to be established. For the convenience of usage, database has been set on the localhost (own laptop). Furthermore the code for interactions (i.e. saving, reading, deleting, transforming data) with the database has been developed.

[4] As the integrated development environment and graphical user interface application to use MongoDB, Studio3T has been chosen.

Data cleaning process focused on filling empty values (not a numbers – "NaNs") with zeros, converting the units and a format of data. As the format used in MongoDB is JSON and the desired type of object to be used with Python libraries in this project was ndarray array (NumPy's array). Lastly, having the data in the right format and units, the normalization method could be applied. This step is vital for machine learning algorithm, as after normalization all data will be within the same magnitude (between 0 and 1), what improves the performance of models. The structure of code and methods used are presented on the following Figure 10.

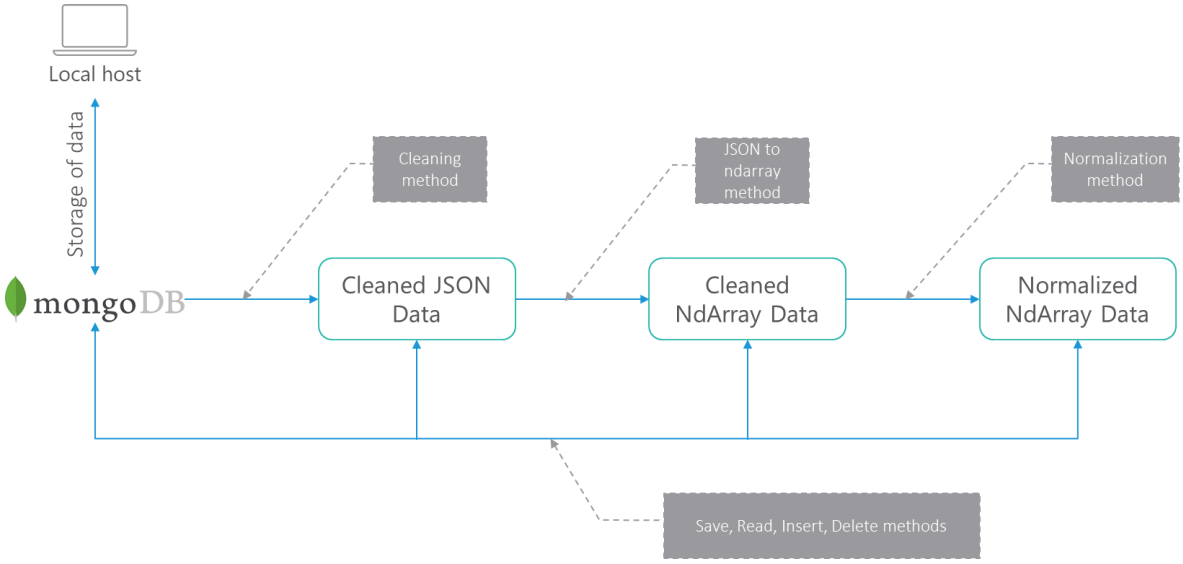

Figure 10 Data cleaning and preparation process.

3.4.2 Exploratory Data Analysis

EDA allows to understand data and relations between variables before introducing it to machine learning algorithms. The description of data is included below:

1. Data consists, depending on the counter installed in BTS, from five to six variables: total electricity consumption (kWh), AC electricity consumption (kWh), Interior Temperature (°C), Exterior Temperature (°C), Speed of the fans (% of maximal value), and AC status (categorical data - 0, when AC is off or 1 when it is on).
2. Data is sampled in 15 minutes intervals.
3. There are 426 BTS-es located in Spain (mainland and islands).

With the functionality of pandas library "describe" method, one can observe the statistical summary for given dataset. It is worth mentioning that these analysis can be done both for normalized and not-normalized data.

The summary consists of following metrics:

- Number of rows,
- Mean value,
- Standard deviation,
- Minimum value,
- Maximum value,
- Percentiles (25%, 50%, 75%).

Furthermore, from the visual perspective, two particular plots are extremely helpful to illustrate the relations between variables – correlation matrix and pair plot charts. First one displays how strongly correlated features are, whereas the second one simply plots features against each other and includes the histograms or probability density charts.

All of that gives a good numerical and visual understanding about the type, distribution and variability of data and serves as an input knowledge to the feature engineering process.

3.4.3 Feature Engineering

Not all of the features might be important from a machine learning perspective and, even if they are, not all of them might be needed to attain a level of confidence in model predictions. The other aspect is the fact that data with high dimensionality (many variables and samples) stands as a serious quest for machine learning algorithms. This phenomenon is widely known as the curse of dimensionality [13] –the model tends to overfit and performance decreases with the high number of features. To overcome this challenge, some techniques have been devised, including feature selection, dimensionality reduction, and feature extraction. The first one is most commonly used [13]. The method concentrates on creating a ranking of features, based on certain criteria. On the account of that, together with business domain knowledge, one can select the set of features with the highest score as this set would provide the best accuracy and performance.

There can be distinguished three main types of feature selection:

1. Filter Methods

Filter methods select variables regardless of the model. The features are ranked based on the statistical indicators including, correlation coefficients, variance and coefficients of variance. The ranked features then provide a list to make a decision of keeping or removing features based on score. However, this

method does not provide information how features perform on a model and which subset of features is the most optimal one.

2. Wrapper Methods

Wrapper methods consider a set of features to find the best subset of features for a modelling problem. This method can be seen as a search problem, where different combinations of features are tested against performance criteria and compared with other combinations. Predictive model is used to evaluate the different sets of features and an accuracy metric is used to score the set of features. The drawbacks of this method are high computational effort, slow speed, and the fact that it can be only applied to supervised learning (labelled data required).

3. Embedded Methods

Embedded methods are more efficient versions of wrapper methods. Penalty factor to the decision criteria of the model to bias the model to lower complexity is included. The methods have an aim to find a balance between the complexity and accuracy of the model. Similarly to Wrapper Method, it is only applicable to supervised learning problem.

One can easily observe that for the case of unsupervised learning only filter methods can be applied. Additionally, it is vital to understand that the aim of feature selection for clustering, which is unsupervised learning problem, is to find the smallest feature subset that best uncovers “interesting” results from data by the chosen criterion. Conversely to supervised learning, in unsupervised one someone has to define what “interesting” and “natural” mean. Apart from empirical and industry knowledge, there are certain statistical criteria that can help with the decision process in terms of data quality and potential to generate good clustering results. Taking all of that into consideration, the two indicators have been used for feature selection process:

A. Statistical Method Score

The score is calculated as average of normalized scores by features in Maximal Compression Index, Variance and Coefficient of Variance.

B. Laplacian Score (LS)

It studies how to select features according to the structures of the graph. The better given data point of one variable fits in the graph the better quality given variables achieves. LS is very effective and efficient with respect to the data size. The most time consuming in LS is constructing the similarity matrix.

The full description of the methods is presented in the Annex A - Feature Engineering.

3.4.4 Clustering

As mentioned in the problem framing, there are no labels (target variables) available to train the model on. Thus as a solution clustering has been selected for discovery of clusters (behaviours) based on the input data. Depending on the clustering method used, there are different decision criteria upon which the algorithm decides to which cluster assign given sample. In general, these criterion can be divided into distance-based – algorithm assigns sample to the cluster that is situated in the closest proximity and probability density ones, that assign data points based on the highest probability of given sample to be clustered to given cluster.

In this thesis, the following clustering algorithms have been applied and tested:

Table 2 A brief characteristics of clustering algorithms applied in the master thesis.

Algorithm	Best use cases	Scalability	Clustering Metric	Speed
K-Means	General-purpose, even cluster size, flat geometry, not too many clusters	Very large number of samples Medium number of clusters	Distances between points	Fast
Hierarchical Agglomerative	Many clusters, possibly connectivity constraints, outlier detection	Large number of samples and clusters	Pairwise distance	Medium
Birch	Large dataset, outlier removal, data reduction	Large number of samples and clusters	Euclidean distance between points	Fast
DBSCAN	Non-flat geometry, uneven cluster sizes, outlier detection	Very large number of samples Medium number of clusters	Distances between nearest points	Medium
Mean-Shift	Many clusters, uneven cluster size, non-flat geometry	Not scalable with number of samples	Distances between points	Very Slow
Expectation-Maximization	Flat geometry, good for density estimation	Not scalable	Mahalanobis distances to centers	Fast

The clustering process can be observed on the schema presented on Figure 11. The final results are presented on interactive charts generated with HighCharts library and the quality of clustering is displayed together with frequency of occurrence of each cluster.

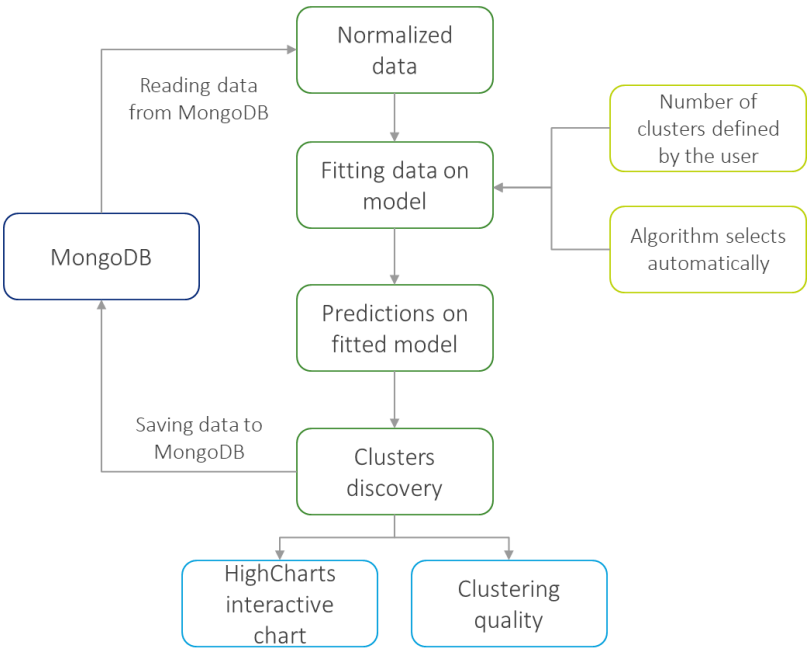

Figure 11 Clustering process schema.

As the quality measure, the algorithms are compared in terms of value of clustering statistical perspective the Silhouette Coefficient and Calinski-Harabaz Index have been measured. Both indicators asses the quality based on separability between clusters and distances of points within the cluster. The longer description is presented in the Table 3 [14].

Table 3 Description of metrics to evaluate the statistical quality of clustering.

Metric	Description	Drawbacks	Desirable value
Silhouette Coefficient	The score is bounded between -1 for incorrect clustering and +1 for highly dense clustering. Scores around zero indicate overlapping clusters.	The Silhouette Coefficient is generally higher for convex clusters than other concepts of clusters, such as density based clusters like those obtained through DBSCAN.	Higher value relates to a model with better defined clusters
Calinski-Harabaz Index	The score is higher when clusters are dense and well separated, which relates to a standard concept of a cluster. The values are not limited to certain boundaries unlikely in the case of Silhouette Coefficient.	The Calinski-Harabaz index is generally higher for convex clusters than other concepts of clusters, such as density based clusters like those obtained through DBSCAN.	Higher value relates to a model with better defined clusters

3.4.5 Labelling

The result of clustering process is the data assigned to different clusters. In simple words, each sample is allocated to a cluster. Nonetheless, this allocation does not name the clusters properly. It is a responsibility of the user to select the names for the discovered clusters. The labelling process takes advantage of three factors:

1. Expert knowledge about the domain and device.
2. Visual interpretation of results for each algorithm.
3. Statistical metrics reflecting the quality of clustering.

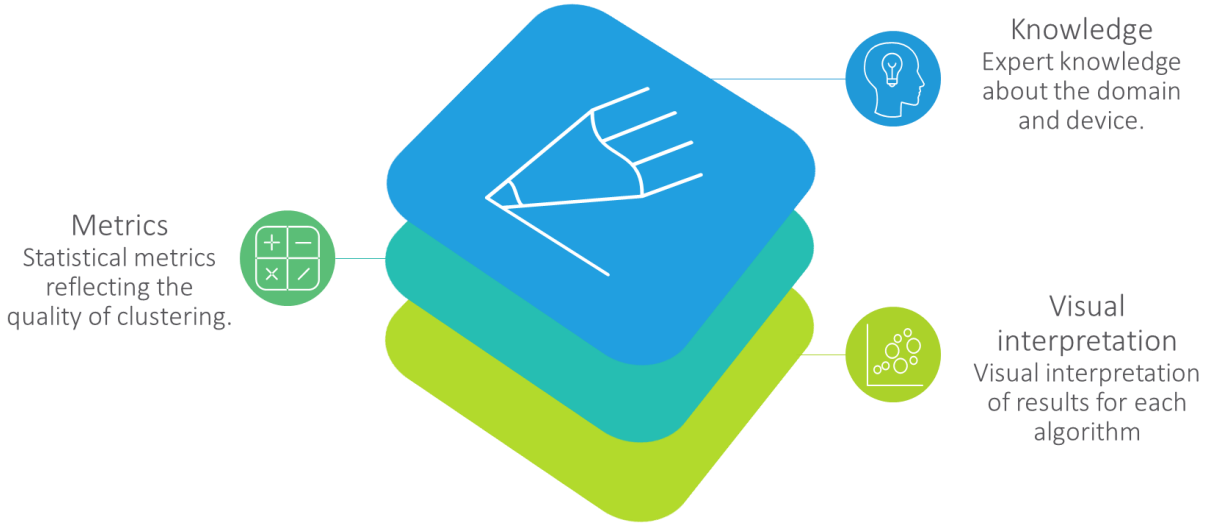

Figure 12 The decision-making criteria to name the labels.

Based on these three inputs, user is able to make a decision and name the clusters accordingly to preferences.

3.4.6 Classification

Once the clusters have been given new names, the labelled data becomes available. Thus the process can be advanced to supervised learning, where the classification model is trained based on the target data and later on can be used for predictions with new data.

In this thesis, the classification algorithms displayed in Table 4 has been applied.

Table 4 Classification algorithms – brief characteristics.

Algorithm	Brief characteristics
Logistic Regression	Logistic regression is intended for two-class or binary classification problems, can become unstable when the classes are well separated
Decision Tree	Robust to outliers, scalable, and able to naturally model non-linear decision boundaries thanks to their hierarchical structure
KNN	Very fast and robust, sensitive to outliers or irrelevant data. Performance depends on the number of dimensions
Linear Discriminant Analysis	Modified Logistic Regression for more than two classes problems and where the problem is linear.
Gaussian NB	Simple and well performing, They are easy to implement and can scale with your dataset. However it suffers from lack of optimization.
SVC	Fairly robust against overfitting, especially in high-dimensional space. However, SVM's are memory intensive, trickier to tune due to the importance of picking the right kernel, and don't scale well to larger datasets.

Hence, in order to run classification algorithm, one has to split the data into train and test subset. The first one serves as material for a model to learn, whereas second one acts as validator of prediction. This split of data into two subsets can have negative effects on the predictive accuracy of model, as data is divided randomly, only specifying the share of data each group has. These negative effects are call overfitting and underfitting. In the first case, a model was “over-trained” and fits too closely to the training dataset. This happens usually when the complexity of a model is high. As a result, the model will perform very well on training data but not very accurately on test data. This happens because model is not generalized for a wide spectrum of behaviours. In the second case, a model does not fit well on training data and is not capable of capturing the trends and patterns in the data. Thereafter, it can be translated on a poor generalization ability to new data. Conversely to overfitting, it usually occurs when the volume of data is small [15].

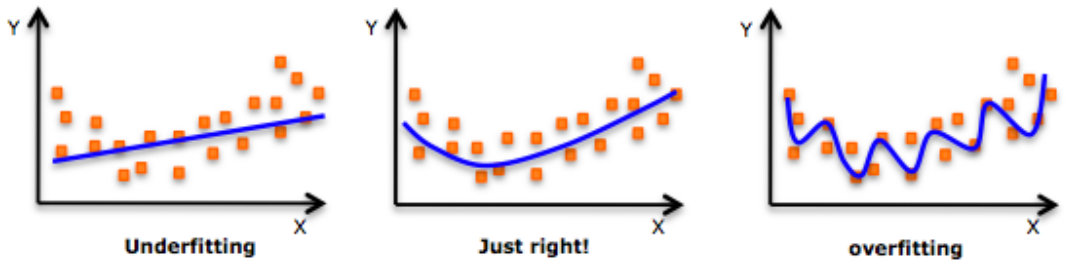

Figure 13 The illustration of underfitting, overfitting and an adequate fitting of the model on the data [15].

Henceforward, to address the issues mentioned above, the model in this thesis includes K-fold cross-validation process. It assures the proper training of model with the lowest possible risk to over or underfit

the model. In this method, data is split into k different subsets. k-1 folds is used to train data, whereas the last subset is left as test data [15].

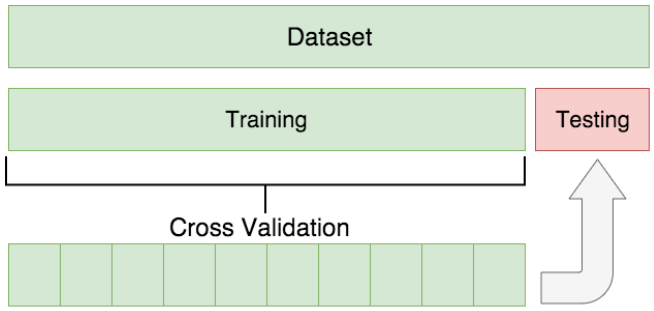

Figure 14 The split of dataset for cross-validation process [15].

After running the process for all k-1 folds the model is averaged against all folds. This combined version of the model that was trained on k-1 data subsets is used for predictions with the last test set [15].

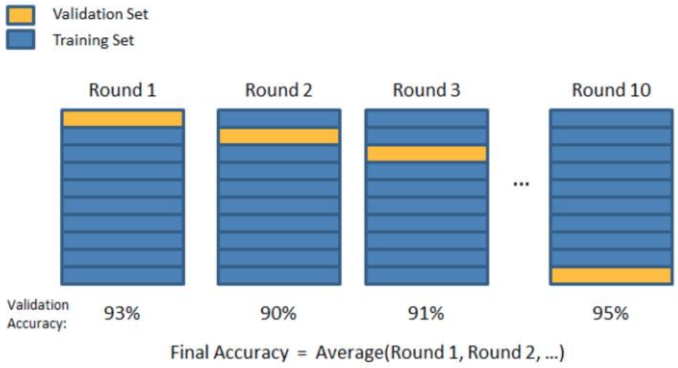

Figure 15 An example of 10-fold cross-validation process [15].

Together with cross-validation a grid-search technique is applied in the program to unravel the best combination of parameters for each algorithm. The method runs classification with cross-validation for all hyperparameters defined by the user. It returns the best algorithm and optimized values of parameters. These two techniques allow to increase the performance and accuracy of models.

The schema of all process required in classification stage are presented on Figure 16. The final results are presented on interactive charts generated with HighCharts library together with frequency of occurrence of each label.

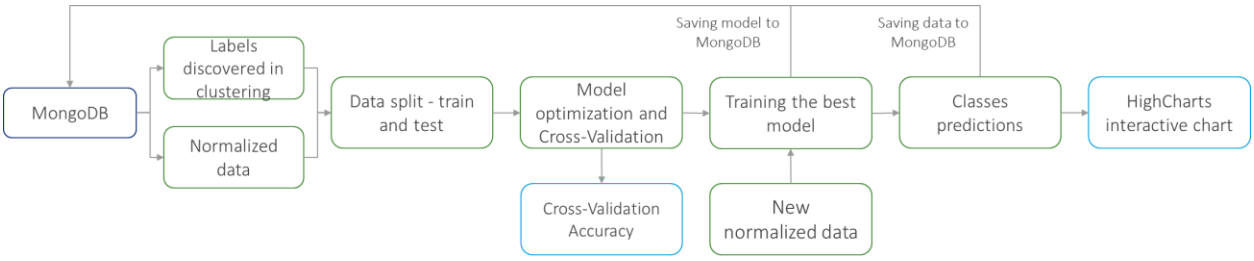

Figure 16 The schema of classification process with Grid Search Cross Validation process.

Lastly, after the model, that is optimized and trained, is being saved to MongoDB. It enables instant predictions without a necessity run optimization and training each time user wants to make forecast on a new data.

3.4.7 Pattern Recognition

Final stage of the modelling process is the pattern recognition service. As its foundation it holds the detection of the patterns that are leading to the label defined by the user. This recognition is based on static aggregation of different sequences of labels with the last one being the behaviour user wants to spot.

The process is displayed on the Figure 17.

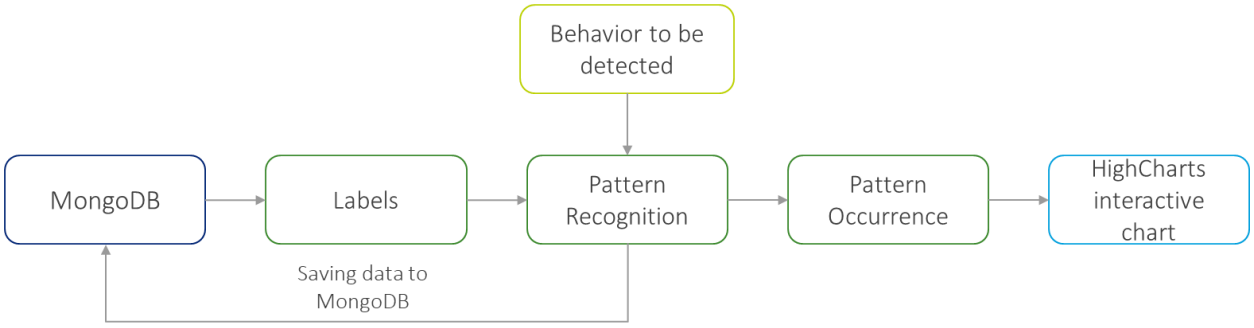

Figure 17 The schema of pattern recognition process.

The process itself consists of labels being imported from database, which are later passed to the pattern recognition algorithm. It creates the "database" of all patterns with selected time window. Furthermore, user inputs an information about a behaviour that he or she desired to be detected. Algorithm detect all patterns leading to selected label. Finally, the results are displayed on interactive charts generated with HighCharts[5] library together with frequency of occurrence of each pattern.

[5] In the final product, charts plotting is moved to the frontend process.

3.5 Platform Implementation

The machine learning program has been created with the intention to be implemented on Wattabit's platform. Since user does not have to see all operations and it is even more desired not to overwhelm her or him, the application is incorporated as a backend process. Graphical User Interface (GUI) placed on the frontend will serve as a medium to allow user to interact with process and make inputs. These inputs and actions are transferred as request to Wattabit's API which calls required processes. After the results from given process are ready, they are being obtained by API and sent to the frontend. Where the process of plotting with HighCharts library is done to display the outcomes to the user. The simplified schema of the program's implementation is presented on Figure 18.

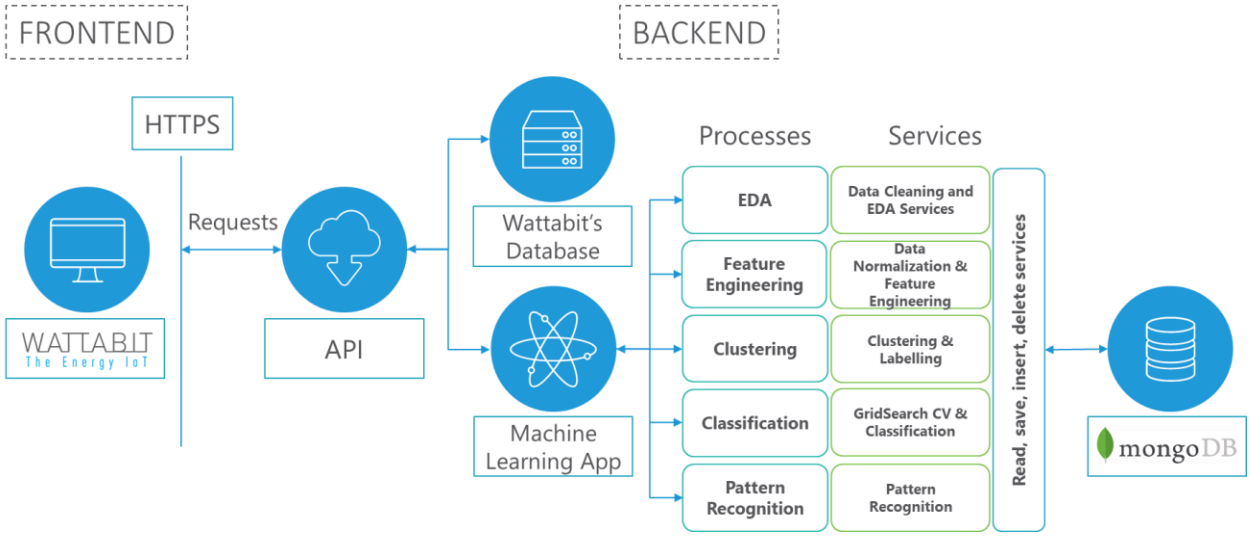

Figure 18 The schema of machine learning application on the Wattabit's platform.

4 Results

4.1 Exploratory Data Analysis

4.1.1 Data Description

The 2017 quarter hourly data measured for one of the BTS-es located in Spain has been imported from Wattbit's database and analysed. The following set of variables was used:

- AC electricity consumption (kWh)
- Total electricity consumption of BTS (kWh)
- AC status (ON/OFF – 1/0)
- Speed of the fans of Free Cooling System (0-100%)
- Exterior Temperature (°C)
- Interior Temperature (°C)

Due to the fact that in some locations Free Cooling System operates differently during the day and the night[6], a variable specifying the period of the day ("day" and "night") has been added.

The following variables characterizes the modes of the Free Cooling System Operation:

- Maximum allowed interior temperature 36 °C[7],
- Day speed max: 50%,
- Night speed max: 30%

All states of operation which surpass these limits can be perceived as emergency or failure mode of operation.

[6] The speed of the fans is reduced in the night time due to the noise issues experienced with the higher speeds.
[7] Over this value the emergency mode is triggered and AC is helping to reduce the interior temperature.

4.1.2 Data Summary

With the use of pandas library in Python, the following data summary has been created:

Table 5 Statistical summary of the data.

Name	Total [kWh]	AC [kWh]	T Interior [C]	T Exterior [C]	Fans [%]	AC status[0/1]
count[8]	35040	35040	35040	35040	35040	35040
mean	0.647	0.017	30.539	19.571	27.826	-[9]
std	0.161	0.071	6.343	7.895	13.679	-
min	0	0	0	0	0	-
$Q_{25\%}$[10]	0.6	0	28.5	13.6	22.25	-
$Q_{50\%}$	0.6	0	32.3	20.1	29.5	-
$Q_{75\%}$	0.7	0	33.5	26.2	33.5	-
max	1.5	0.7	39.9	38.1	100	-

Additionally, based on the data the following annual values have been calculated:

- Total base consumption[11]: 31611.34 kWh.
- Total electricity consumption of BTS in 2017: 22683.1 kWh.
- Total AC consumption: 602.9 kWh (2.66% of total consumption)
- Total electricity savings coming from FCS implementation in 2017: 8928.24 kWh
- Percentage of time AC was running: 7.70%
- Percentage of time FCS was running: 88.62%
- Total CO_2 emissions: ~6.5 t CO_2 (28% reduction)
- Total CO_2 savings: ~2.5 t CO_2

4.1.2.1 Insights

The most useful insights coming out of Table 5 are the general information, allowing to see the magnitude of total and AC consumption. Moreover, the percentiles give an idea which values occur most frequently (in 75% of instances) for fans operation – speed is lower 33.5%, interior temperatures are lower than 33.5 °C and electricity consumption being less than 0.7 kWh.

[8] It describes the number of rows being not empty, which is equal the number of time stamps registered.
[9] There are no metrics for AC status because it is categorical type of data.
[10] 25% percentile of data distribution.
[11] Estimated value of electricity consumption without FCS.

Additionally, the relative standard deviation draws an attention to the variance in data, highlighting AC consumption as the feature with highest coefficient of variance. Furthermore, taking a look at the annual values one can easily observe that the saving coming from FCS implementation are enormous. The total electricity consumption reduction reaches the level of 28%, which translates into great monetary and CO_2 emission savings.

Taking into consideration the price of electricity and CO_2 emission factor from Chapter 6 Environment impact, a final reductions are respectively achieved at the level of € 540.6[12] and 2.5 t CO_2 annually. The distribution and variance of data can be noted by looking at the standard deviation and quantiles. One can see that the highest deviation in the absolute values was obtained for the fan speed and the temperatures. However looking at the relative standard deviation, AC consumption is characterized by the highest coefficient of variance.

4.1.3 Correlation Matrix

In order to determine the strengths of relationships between variables the correlation matrix has been developed. The data is presented in the heat map form with the correlation coefficients on Figure 19.

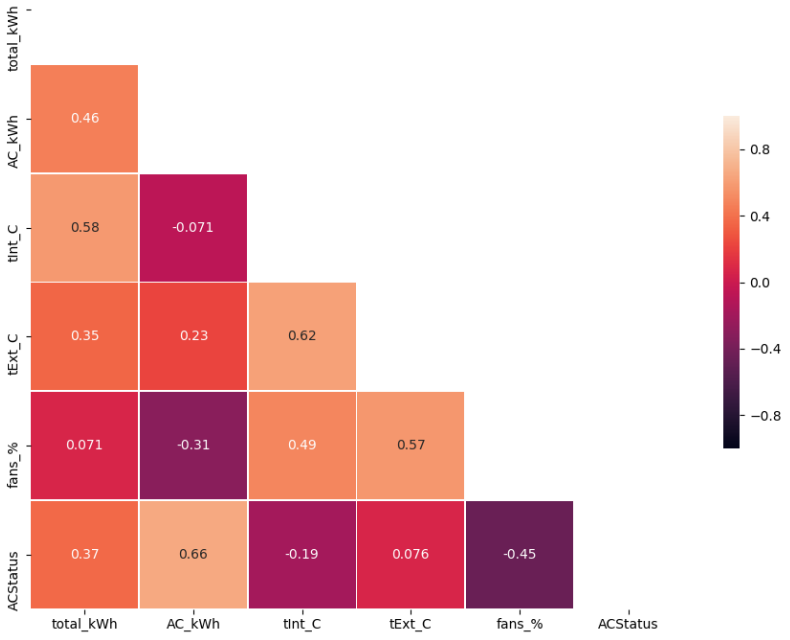

Figure 19 Heat map of correlation coefficients.

[12] Taking into consideration the average price of electricity from 2017 [19].

4.1.3.1 Insights

Based on the correlation matrixes and coefficients, one can have a complete picture about the dependencies or independencies between the variables. Hence, the strongest positive correlation exists between AC status and its consumption. Going along that, the notable positive correlation can be also seen between exterior and interior temperatures, fans speed with exterior temperature, and total consumption with interior temperature.

On the contrary, the most prominent negative correlation can be noted for speed of the fans and AC status. These relations will serve later on in the process of feature selection for machine learning program.

4.1.4 Relations between variables

The relationships between the variables was assessed in the pairs of two, checking the behaviour of one variable to the other. The cumulative chart, consist of several small scatter charts between two features, with the histograms being plotted for each feature on the diagonal.

Furthermore, the two cases are distinguished, Figure 20 presents the division between day and night operation, whereas Figure 21 displays the situations when the system was running in the emergency mode[13].

[13] Value 1 indicates emergency mode being active, and 0 the opposite one.

Figure 20 Pair plots with day and night operation division.

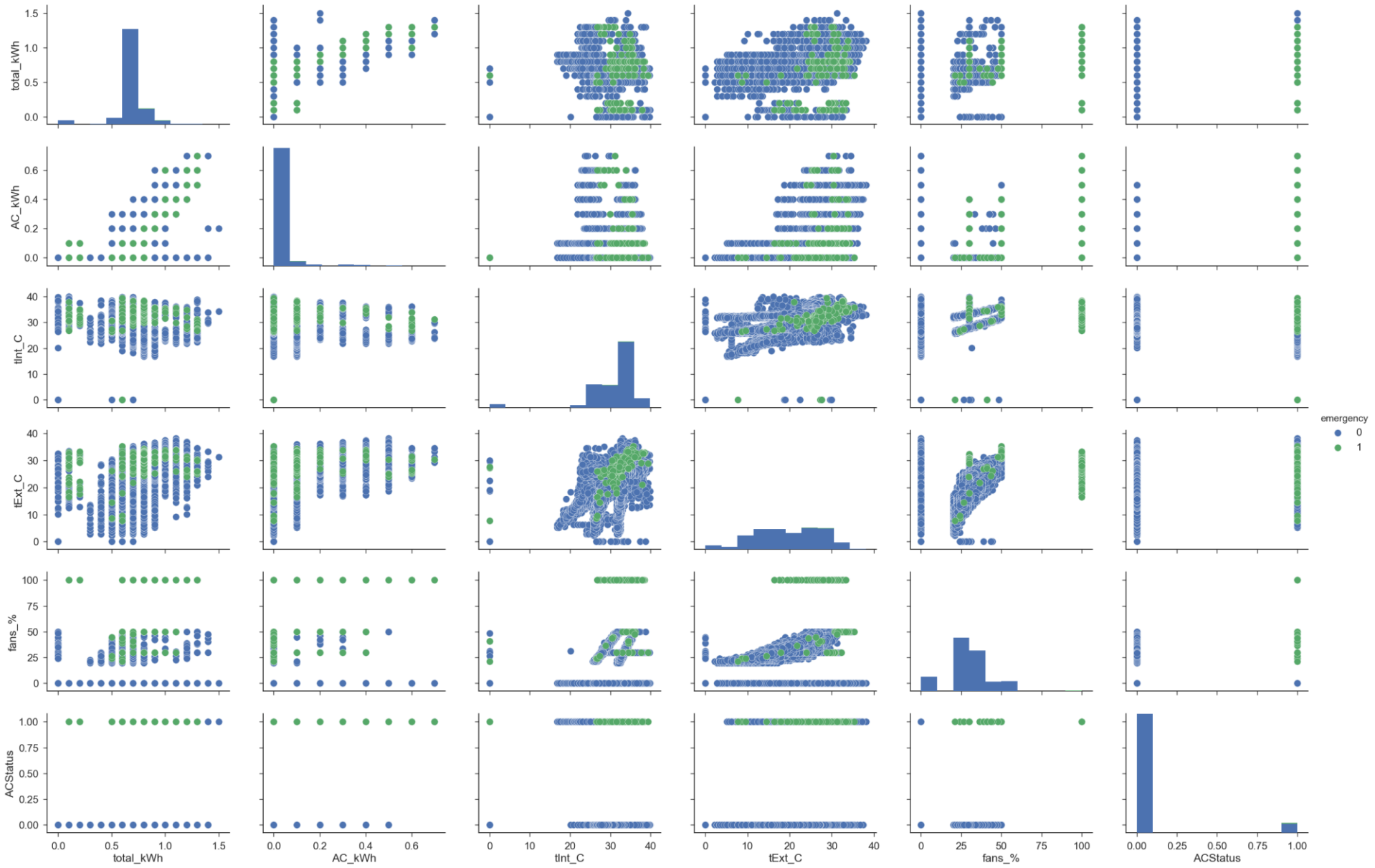

Figure 21 Pair plots with emergency mode marked.

4.1.4.1 Insights

First of all, the pair plots display the operation range of FCS, which is in the function of interior and exterior temperature. The histograms give insights and supplement the percentiles description from Data Summary, how the data is being distributed and which values are occurring most frequently. The linear relations can be observed for: AC consumption and exterior temperature with total consumption. The latter part also illustrates at which values the emergency mode had to be activated. For analysed BTS it was 202 times in 2017, highlighting the importance of proper detection of this pattern.

4.2 Feature Selection

The purpose of feature selection process is to choose the features which have the greatest explanatory power and fit aptly to the clustering algorithms. In order to accomplish that, two methods previously described in the Subchapter 3.4.3, have been applied. The results are visible below.

4.2.1 Statistical approach for feature selection

The statistical method takes into consideration three indicators, Maximum Compression Index, Variance and Relative Standard Deviation (CoV – Coefficient of Variance). After the results for each metric are obtained they are normalized to allow to do an unbiased comparison and enable the final metric to have the same magnitude.

As a result, the ranking is obtained and the features are positioned with the best feature having the highest average score, as it reflects the ranks in the three indicators rankings mentioned above. The list of rankings for three indicators mentioned above can be observed in Table 6.

Table 6 Statistical feature selection ranking.

Feature Name	Variance Score	CoV Score	MCI Score	Average Score	Final Ranking
AC Status	1.0000	0.8330	1.0000	0.9443	1
Exterior Temperature	0.5375	0.0501	0.7952	0.4609	2
Interior Temperature	0.2472	0.0000	0.4209	0.2227	4
Speed of fans	0.1394	0.0727	0.2797	0.1639	5
Total Consumption	0.0201	0.0103	0.0595	0.0300	6
AC Consumption	0.0000	1.0000	0.0000	0.3333	3

4.2.2 Theory of graphs approach for feature selection

Laplacian score allows to create an importance ranking. The smaller the Laplacian score is, the more important the feature is. The outcome of analysis is presented in the Table 7.

Table 7 Feature selection graph based on the Laplacian Score

Feature Name	Ranking	Score
AC Status	1	0
Speed of fans	2	0.000627
Exterior Temperature	3	0.001083
AC Consumption	4	0.00153
Interior Temperature	5	0.002342
Total Consumption	6	0.003065

4.2.3 Summary of results

One can easily see that the two rankings do not provide entirely the same insights. Hence the decision, in the case of problem with small number of feature, should be carefully evaluated. As all features may be useful to unravel various behaviours.

Since the difference in the rankings and the scores of each feature does not allow to make an obvious selection, therefore, in this case, the decision was made to keep the original set of features, not removing any of them. However, the method is extremely useful when the dataset has more variables. Lastly, considering the computational time the statistical ranking is faster (few seconds) to obtain. On the other hand, it takes over 150 seconds to calculate the Laplacian Score.

4.3 Clustering

After the process of selecting the most adequate features, models previously described in the Subchapter 3.4.4 are fed with the normalized data. The tests included the sensitivity analysis changing the number of clusters[14]. For each iteration and algorithm the computational time and the clustering quality scores were calculated.

4.3.1 Preface

In the paragraph below, the results for the best quality results assessed with visual interpretation and support of the statistical metrics are presented. The structure of the chart that presents the results can be seen on Figure 22. There are 6 variables plotted on one chart with the plot bands in the background, indicating the cluster to which given timestamp has been clustered to.

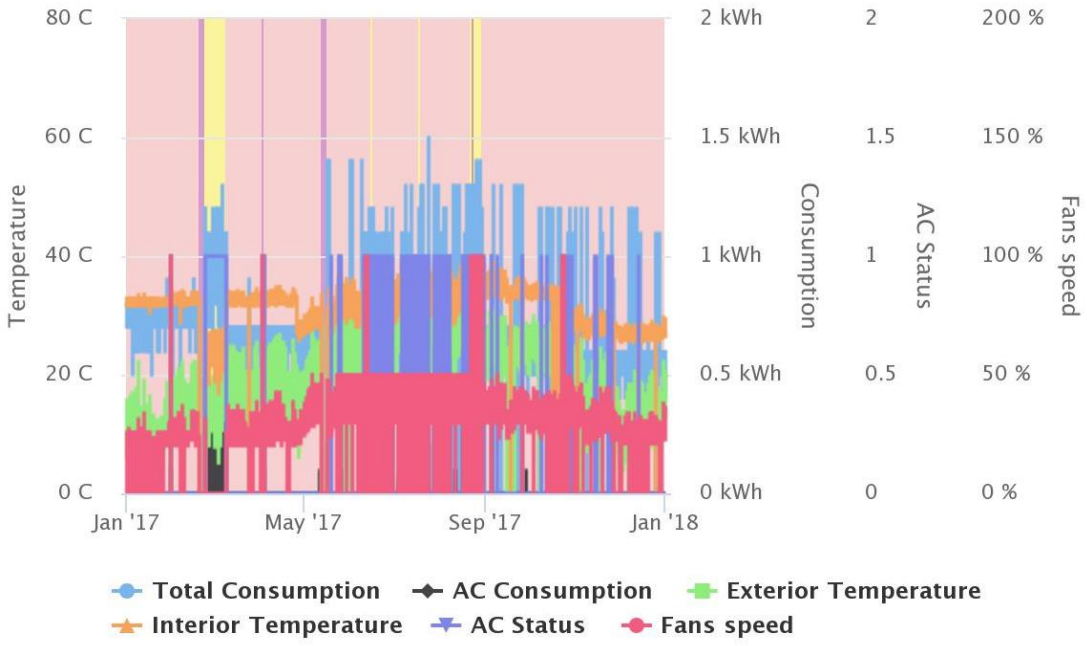

Figure 22 Example of the chart as the results of the clustering using DBSCAN.

Since, this way of presenting results does not give the reader any idea how to interpret them, the charts in the picture format will not be displayed in this chapter. However, to address that issue the direct links to interactive HTML[15] charts in bigger format have been enclosed. Additionally, all screenshots of the charts are included in Annex C – Results charts.

[14] Excluding DBSCAN and Mean Shift algorithms which calculate automatically a number of clusters.
[15] In order to open the link, please press CTRL and the link.

A. K-Means

According to the statistical metrics, the highest quality was observed for clustering with 9 clusters. Nonetheless, from the visual experience, a clustering with 7 clusters carries the same information, without adding useless noise. The separation of clusters is at decent level, seizing the most important and abnormal behaviours. The outcome implies that the algorithm is able to recognize the movement (increase and correlation) of fans speed and temperatures. Nonetheless, some clusters (6,2,0) could be described as the same normal behaviour. On the other hand, an occasional misclustering is also noticeable.

a. Chart: K-Means Clustering with 7 clusters

b. Clusters Occurrence

As mentioned above, clusters 6, 2, and 0 indicates the normal behaviour. Whereas, cluster 3 shows the lost connection mode. The emergency mode is included in the cluster number 5 and operation mode with only AC on is presented as cluster 4. The rest of behaviours is a mixture of different ranges of parameters that do not provide any fruitful insights.

Table 8 The clusters occurrence for K-Means with 7 clusters.

Name	Occurrence
Cluster 6	31.51%
Cluster 2	28.75%
Cluster 3	2.95%
Cluster 5	0.51%
Cluster 1	4.07%
Cluster 4	3.12%
Cluster 0	29.09%

B. K-Means with PCA[16]

Using Principal Component Analysis to compress data with the K-Means algorithms, the obtained results are to the case with normal set of data. However the noise in the data and the misclustering are very significant. Despite the fact that from statistical point of view, solution with 9 clusters provides the highest score, from visual interpretation clustering with 8 clusters was identified as the best one.

[16] The PCA method compresses the original set of features into selected by user number of dimensions (in our case, two dimensions were chosen) and feeds the model with compressed data.

a. Chart: K-Means PCA with 8 clusters

b. Clusters Occurrence

Clusters 4, 1, 6, 3 all represent the normal behaviour of the device. Similarly to the K-Means with normal data the emergency and AC on modes are also identified. However the results of other clusters are very noisy and not easy to evaluate.

Table 9 The clusters occurrence for K-Means PCA with 8 clusters.

Name	Occurrence
Cluster 4	19.15%
Cluster 1	25.19%
Cluster 6	23.57%
Cluster 2	2.94%
Cluster 7	1.55%
Cluster 5	3.86%
Cluster 0	2.29%
Cluster 3	21.45%

C. Hierarchical Agglomerative

In the case of hierarchical agglomerative clustering, the most optimal option was identified to be 8 clusters. The option with less groups does not provide the complete information about the behaviours, whereas the ones with higher number bring noise and overlapping clusters.

a. Chart: Agglomerative clustering with 7 clusters

b. Clusters Occurrence

The same situation as in the other cases, clusters 1, 0, and 5 indicates the normal behaviour and can be merged. The algorithm detects also the other modes of operations. All in all, the noise is at moderate level.

Table 10 The clusters occurrence for hierarchical agglomerative clustering with 7 clusters.

Name	Occurrence
Cluster 1	26.85%
Cluster 0	43.14%
Cluster 5	19.37%
Cluster 3	2.94%
Cluster 6	0.43%
Cluster 2	3.42%
Cluster 4	3.85%

D. Hierarchical Birch

In the case of birch clustering, the best results turned out to be for 6 clusters. The characteristics of this method apart from standard types of behaviour provides also information about the peaks in the electricity consumption. In general, the results are very good compared to for instance hierarchical agglomerative clustering.

a. Chart: Birch clustering with 6 clusters
b. Clusters Occurrence

Apart from cluster 1 and 3 being the normal behaviour, cluster 4 provides an interesting information about the peaks in the electricity consumption. The rest of the clusters are separated with a decent quality.

Table 11 The clusters occurrence for hierarchical birch clustering with 6 clusters.

Name	Occurrence
Cluster 1	43.25%
Cluster 3	49.05%
Cluster 2	0.37%
Cluster 0	6.44%
Cluster 4	0.83%
Cluster 5	0.05%

E. Expectation-Maximization Gaussian Mixture (EM)

With the increase of number of clusters, the results become to carry more noise and misclustering. Therefore, the most optimal outcome with clean results is a clustering with 6 clusters.

a. Chart: Expectation-Maximization Gaussian Mixture clustering with 6 clusters
b. Clusters Occurrence

Cluster 3 seems to be the most interesting one as it reflects the sudden peaks or drops (anomalies). Other than that the clusters bring the same information as in the other methods.

Table 12 The clusters occurrence for Expectation-Maximization Gaussian Mixture clustering with 6 clusters.

Name	Occurrence
Cluster 4	36.88%
Cluster 3	2.79%
Cluster 0	49.70%
Cluster 1	2.94%
Cluster 2	0.61%
Cluster 5	7.09%

F. DBSCAN

DBSCAN algorithm clusters the data without the predefined number of clusters. Instead, it assigns the data points to the clusters until it reaches the optimum of objective function. This methodology, implies the number of clusters created. The results produced by this method provide user with the more general structure, focusing on general types of behaviour but also detecting the outliers.

a. Chart: DBSCAN Clustering
b. Clusters Occurrence

One can see that DBSCAN algorithm clustered the data in 6 clusters. The cluster -1 stands for the outliers, which displays the situation when the emergency mode was on and no interior temperature was measured. The general behaviour (Cluster 0) was in almost 90% of the examples. The cluster 1 indicates the loss of communication (no data was recorded). Cluster 2 highlights the emergency mode of operation. Whereas cluster 3 stands for AC only operation. Lastly, cluster 4 could be accounted as the outlier as well, since it implies the situation where interior temperature is zero. The only difference is that the system is not in the emergency mode then.

Table 13 Cluster occurrence for DBSCAN method.

Name	Occurrence
Cluster 0	89.35%
Cluster 1	2.94%
Cluster 2	0.43%
Cluster 3	7.26%
Cluster 4	0.01%
Cluster -1	0.01%

G. Mean Shift

Similarly to DBSCAN, the Mean Shift algorithm does not need to have predefined number of clusters. Compared to DBSCAN, it uses different criterion to assign the datapoints. The objective function is provided with the band width parameter that has been optimized accordingly to the data passed.

a. Chart: Mean Shift Clustering
b. Clusters Occurrence

Cluster 0 describes most frequently occurring general behaviour, whereas cluster 2 shows the loss of communication cluster. Furthermore, cluster 1 indicates the AC on operation mode. Emergency mode is described by cluster 3. Whereas the rest of the clusters can be clusters occasional behaviour, like interior

temperature not recorded or AC Status on with the bigger difference between external and internal temperatures. The interesting cluster that have been discovered is marked with number 5, which assigns describe the situation when there is big growth in AC consumption and decline in the interior temperature. It can be interpreted as the emergency mode starting to cool down the interior temperature.

Table 14 Cluster occurrence for Mean Shift method.

Name	Occurrence
Cluster 0	89.35%
Cluster 2	2.94%
Cluster 3	0.37%
Cluster 1	6.10%
Cluster 4	1.16%
Cluster 6	0.01%
Cluster 5	0.05%
Cluster 7	0.01%
Cluster 8	0.00%

4.3.2 Comparison of scores and computational time

Not only can the clustering quality measure tell one about the most optimal option, but also the computational time needed to run each of the algorithms. The statistical scores for all options[17] have been shown on Figure 23 and Figure 24.

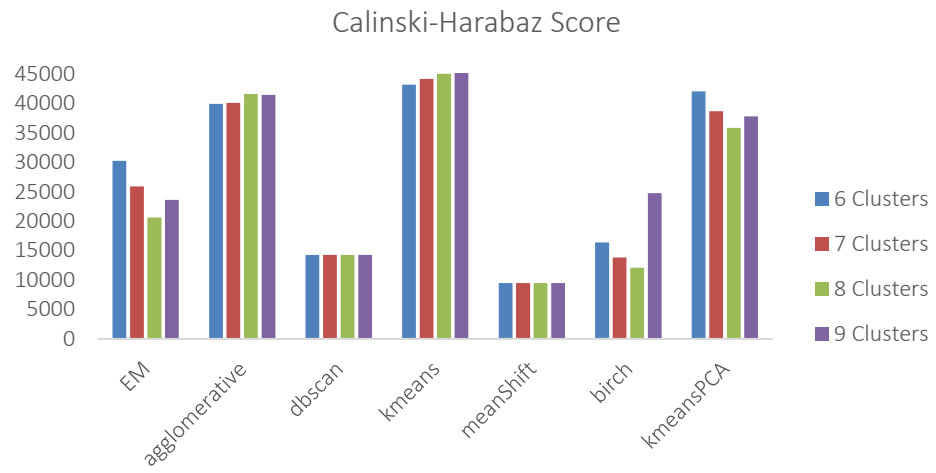

Figure 23 Calinski-Harabaz Score.

[17] For Mean Shift and DBSCAN methods the results are all the same since the algorithms calculating the optimal number of clusters automatically.

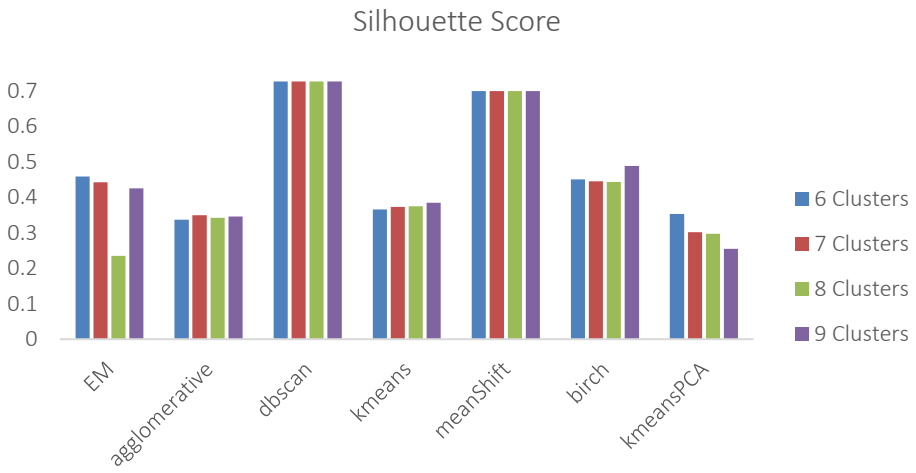

Figure 24 Silhouette Score.

The computational time comparison for each algorithm is presented on Figure 25.

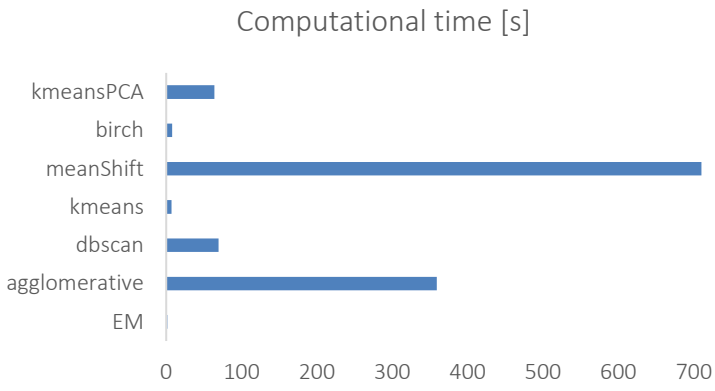

Figure 25 Computational time comparison of all algorithms.

4.3.3 Summary of results

With the increase in the number of clusters, the algorithms tends to produce various results. Even though the statistical score improves, not always does one can obtain better results. Going along that, some of the algorithms, with the greater number of clusters, are able to detect new types of behaviours, however the overall quality of the rest of the clusters is negatively impacted.

The one of the greatest discovery is the detection of the cluster that highlights the situation when despite the fact that emergency mode is switched on, the interior temperature is still rising, surpassing the allowable limits. It can be seen on the following example as the plot band with pink-purple colour.

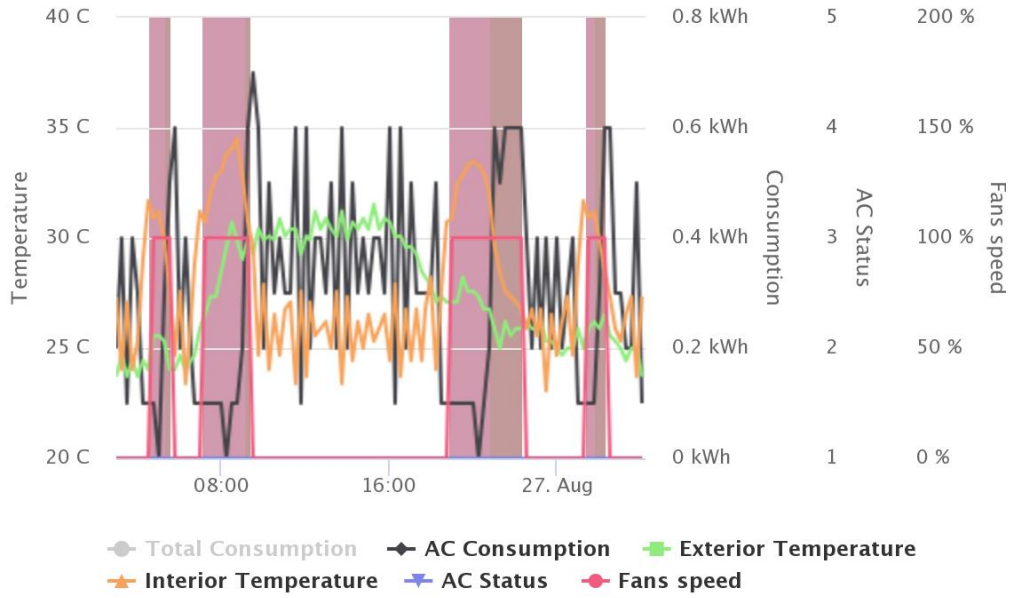

Figure 26 Abnormal behaviour of cooling system.

Secondly, some of the algorithms (EM) are cable of detecting the fault in the system, when even though the AC is switched off, some consumption for cooling purpose is recorded. The situation is presented on Figure 27.

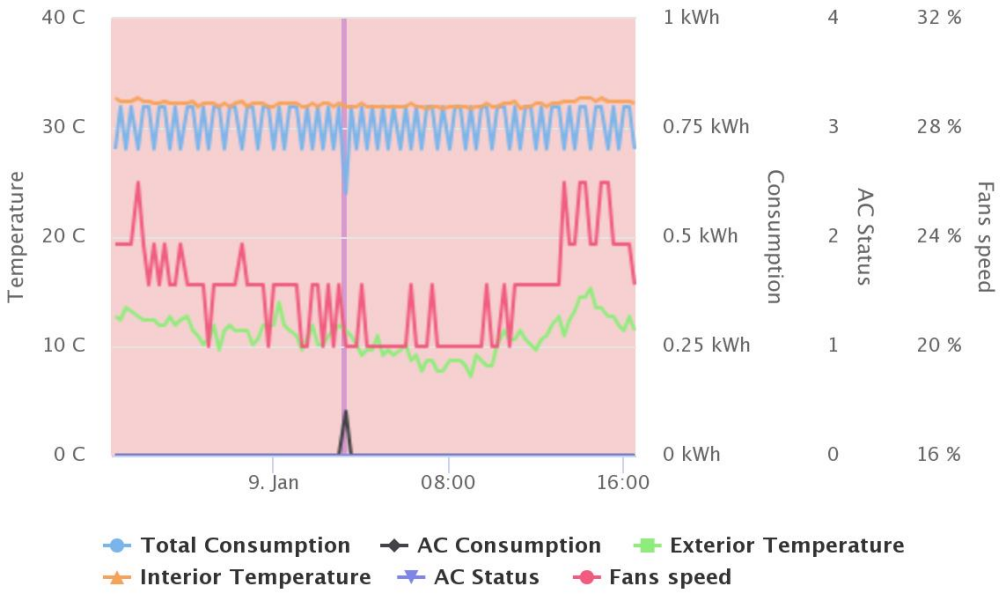

Figure 27 Unusual AC consumption while being switched off.

The other types of behaviour that is of a great value is the lost communication and emergency mode observances. Having the wide spectrum of behaviours detected by clustering algorithms, the process can advance to the labelling stage where one can take advantage of all discoveries done by the models.

4.4 Labelling

After the best results in the clustering phase are obtained, the labelling process can begin. It consists of two phases:

- First phase – merging the results of all algorithms and checking which clusters are overlapping or which ones have been only detected by specific algorithm. It is essential to create new labels for the new type of behaviour.
- Second phase – renaming the clusters with the industry specific nomenclature.

Based on the wide spectrum of behaviour recognized the following results was used to create the labels:

- DBSCAN
  - Cluster 0 as "Normal_Behaviour"
  - Cluster 1 as "Lost_Communication"
  - Cluster 2 as "Emergency_Mode"
  - Cluster 3 as "AC_ON"
  - Cluster 4 as "Outlier"
  - Cluster -1 as "Outlier"
- EM (6 clusters)
  - Cluster 3 registered at 1/9/2017 4:15:00AM as "AC_Consumption_Error"[18]
- Birch (6 clusters)
  - Cluster 2 in August 2017 between 23rd and 28th as "Faulty_Emergency_Mode"

Once the labels are renamed, the chart accessible at the following link is obtained:

Final clustering results with renamed labels

The screenshot of a chart can be observed in Annex C – Results charts.

Finally, the distribution of clusters was recorded and presented in Table 15.

[18] There are more examples of this behaviour however for the sake of simplicity only one date has been specified here.

Table 15 Clusters occurrence for renamed labels.

Name	Occurrence
Cluster Normal_Behaviour	89.29%
Cluster AC_Consumption_Error	0.06%
Cluster Lost_Communication	2.94%
Cluster Emergency_Mode	0.15%
Cluster AC_ON	7.26%
Cluster Outlier	0.03%
Cluster Faulty_Emergency_Mode	0.27%

This data can now be used to train the classification algorithms to make predictions for new devices.

4.5 Classification

In this chapter the results of classification with grid search for hyper parameters optimization and cross-validation are presented.

The grid search algorithm had ran an optimization process to find the best hyper parameters for each model. The description of the subsets of parameters for each classifier is included in the Annex B - Classifiers Optimization.

The 5-fold cross-validation has been applied due to the high volume of the data. The original set was divided into training and testing subsets, with the size of respectively 70% and 30% of initial set.

After obtaining the optimized hyper parameters, the training and validation of models have been introduced. The performance for each model is presented in the Table 16, whereas the computational time is displayed on Figure 28.

Table 16 The comparison of training, test, and computational time results.

Method	Test Score	Train Score	Time [s]
Logistic Regression	0.999619482	0.955984997	344.0040002
K-Neighbors Classifier	0.999238965	0.99853229	19.28200006
Gaussian NB	0.983637747	0.982632094	1.380000114
Linear Discriminant Analysis	0.993055556	0.952804958	8.299000025
Support Vector Machine	0.999619482	0.958506119	679.141
Decision Tree Classifier	0.999238965	0.999011334	4.225999832

The highest score has been achieved by two algorithms: Support Vector Mechanism and Logistic Regression. Thus, for the prediction phase, only one of them was required. The decision was to select Support Vector Mechanism[19], as it provides a slightly better score (decimal points, not visible in the table).

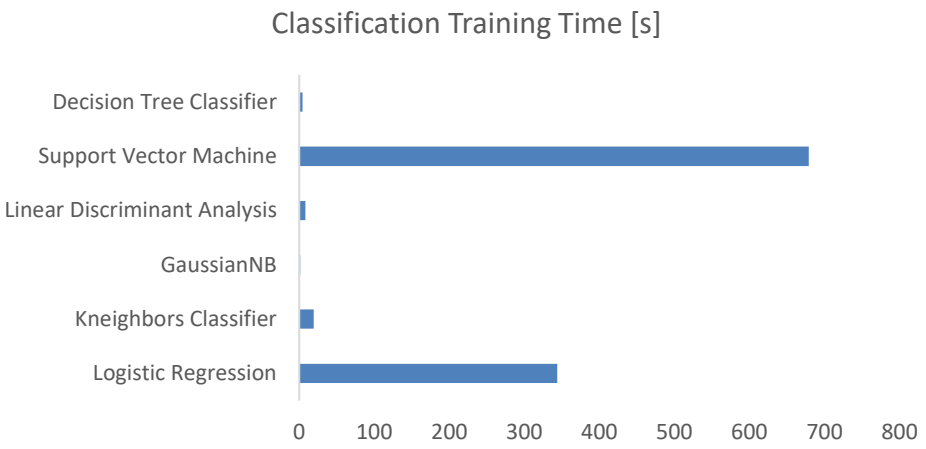

Figure 28 Computational time comparison.

The SVM model, previously trained, is imported from database and the predictions are made. Visually results are very accurate, perfectly matching assigning new datapoints to the correct classes of behavior. The visual representation of the predictions can be seen on the figure included in the hyperlink below.

Support Vector Machine Classification Results

In comparison to clustering results, the chart is enriched with the probability of assigning each data point to given class.

The screenshot of a chart can be found in Annex C – Results charts

[19] Logistic regression model has obtained the same score for test sample but the score for the training subset was lower than for SVM. However, in the case of when the computational time for training the model matters, one can proceed with Logistic Regression as it is twice faster as SVM.

4.6 Pattern Recognition

In the final stage of the project, the patterns are being detected. The algorithm takes as an input the desired behaviour to be recognized and based on the sequence of labels spots various patterns. As the examples three simulations were executed:

1. Detection of AC ON mode with the time window of 1 hour and 15 minutes

   Detection of AC ON mode

   In this case, pattern 0 is the most frequently occurring one. It has lead 146 times to activation of AC system.

2. Detection of Emergency Mode with the time window of 1 hour

   Detection of Emergency Model

   In the case of emergency mode pattern detection, pattern 0 was the most common one and had occurred 26 times.

3. Detection of Faulty Emergency Mode with the time window of 45 minutes

   Detection of Faulty Emergency Mode

   There is only one pattern that has led to the emergency mode not working properly. The pattern was observed 9 times.

The screenshots of charts can be observed in Annex C – Results charts

The results presented consists of patterns named with the integers, where Pattern -3 stands for the time window selected[20], and Pattern -1 indicates the normal behaviour[21].

As one can see, the pattern detection method can unravel very useful insights and serve as a "database" to create personalized alerts for the when the patterns starts occurring. It can bring plenty benefits thanks to early detection and preventive actions.

[20] The time window implies that the data is shifted, thus there is no pattern detection for the first time steps equal to the selected time window.
[21] Not leading to the selected behavior.

5 Business Perspective

Despite the fact that the application has been developed on data coming from BTS, the code and the structure of the program have been designed to be a universal product. On the account of that, a company can include it in the portfolio of services and additionally create an opportunity to generate supplementary benefits related to this application.

5.1 Value Proposition

Figure 29 Value Proposition of the product.

The pattern detection application is offered as tool to discover different behaviours, especially negative anomalies and errors. Thanks to that, the operating expenses related to the prevention of device's failure, declining the need for maintenance check and the operation optimization, can be reduced. Furthermore, as a result of anomalous behaviour detection the maintenance time can be shorten and as frequent inspections in the field are not required. Additionally, application serves as a medium enhancing user's or operator's knowledge about device's behaviour under different conditions.

5.2 Revenues Streams

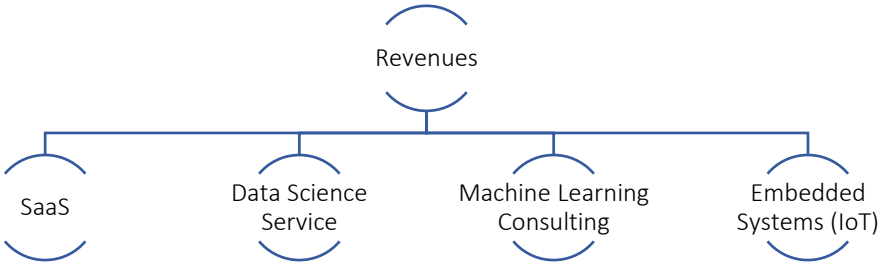

Figure 30 Revenues streams of the product.

The main idea is to offer the functionality to the user in the Software as a Service (SaaS) scheme. Thereafter the main source of revenues would be coming from users fees or subscriptions. Depending on the pricing scheme provided, it could be a pay per device or per a period of time, that would scale up with the number of devices or the length of the contract. The additional revenues could be coming from data science and

consulting services. In the case of the first one, user would be paying for the service of data analysis and model optimization to make a tailor made solution for the user's specific problem. In the latter one, the company could be offering consultancy service to the other companies in the field of machine learning and predictive maintenance, for which a certain remuneration would be asked. Lastly, once the program would be improved it could be implemented in the embedded system under the umbrella of IoT concept to analyse, warn and control the device based on smart data analysis and machine learning algorithms. The revenue would be coming either from the users for selling a turnkey solution (hardware with implemented software) or as a royalty for software in the case of an agreement with the manufacturer and provider of embedded systems.

5.3 Customer Segments

Since the product can be used in different industries, three customer segments can be distinguished:

- Companies with production machines, devices etc. that generate vast amount of data, are vulnerable to abnormal operation, have a significant operating expenses and the cost of failure is immense, e.g. power utilities, oil & gas companies, telecoms, commercial buildings or software companies.
- Companies present in software or IoT business that want to develop the solutions based on machine learning and data analytics.
- Hardware companies that are present in embedded systems market or plan to enter this market and develop "hardware plus software solution".

6 Environment impact

The CO_2 emissions in the data centres are highly correlated with the electricity consumption and by extension the carbon intensity of electricity generation system in the country where data centre is located. Furthermore, the negative impact of electricity consumption related to the carbon dioxide emissions can be mitigated by contracting electricity from a supplier that secures the origin from renewable source. However, in general terms the level of emissions is the factor of the magnitude of consumption and CO_2 emissivity per unit of electricity consumed.

There is also one additional factor that can be perceived as harmful for the environment - the heat dissipation or hot air outlet to the atmosphere that warms the air. However, this impact can be diminished compared to the CO_2 emissions related to electricity. Additionally, the data centres of prominent internet giants like Alphabet, Facebook, Apple and Amazon have already extensively invested in the data centres located in the Nordic countries where the low atmospheric temperature reduces the total need for cooling [16].

Data centres consumed around 1% (194 TWh) of electricity produced globally in 2014. Despite the fact that data centres workload is expected to triple by 2020, the relative growth in the demand is foreseeable at the level of 3% by that time, as a result of efficiency improvements [17].

In the case of telecom data centres analysed in this thesis, the emissivity of electricity generation system in Spain in 2017 was 0.285 t CO_2/MWh [18]. Hence, the impact of one BTS can be assessed at the level of 6.5 t CO_2. Thereafter, for the whole fleet of around 400 BTS-es, the total CO_2 emissions are around 2600 t CO_2 annually.

7 Conclusions

The work accomplished in this thesis proves that the creation of an application, with the use of machine learning algorithms can bring palpable benefits as it is able to unravel different types of behaviours and patterns. The demanding part in the process requires human input to select the best clustering and to name the discovered labels. Hence, an improved solution would be to have already prepared data with labelled behaviours. If the model or group of devices would have that , the whole process would be shorter and faster.

Taking into consideration the quality terms, the combination of different clustering results can bring the best of each algorithm. Thus, the output is more complete and includes the greater spectrum of behaviours, enabling the operator to catch the patterns which could not be spotted without the machine learning model. Furthermore, looking from the product perspective, it is vital to consider the computational time of each algorithm as the user might not want to wait until the model finishes the calculations. As it is hard to rank the clustering outcomes based solely on statistical indicators. On the other hand, in the classification problem it is straightforward as the quality indicator is the accuracy score, that allows ranking the methods. In the case of analysed BTS, the main outcome of this ranking is that all the models performed extremely well, surpassing 98% of accuracy. The last functionality of program brings the best of analysing the sequence of behaviours, leading to a user specified event. Even though it is based on historical data and acts as a static method, it provides user with fruitful insights about types of various patterns and their frequency of occurrence.

Lastly, thanks to the universality of machine learning models, the application can be scaled and applied to any kind of the devices in various industries and commercial sectors[22]. Thereafter, a promising business case[23] can be built upon this product.

[22] Sometimes, there is a need to tune the model or select/extract features adequately, which implies the necessity to have a domain knowledge or to hire a data science service.

[23] As an additional service in the company's portfolio offering SaaS to generate another streams of revenues.

7.1 Further work

The next steps would concentrate on using the application with the stream of real-time data, instead of historical records. Three main functionalities of the upgraded program can be foreseen. Firstly, the detection of the behaviours (labels) in real time. Secondly, the user would have a functionality to create pattern recognition alerts with selected time window and behaviour. Thirdly, the prediction of future label or behaviour shall be introduced based on time series forecasting.

Additionally, the application could be tested with the use of more advanced techniques like: deep learning and ensembled machine learning models to verify and compare the quality of clustering and accuracy of classification. Lastly, with the increase of data volume and dimensions, it is recommendable to migrate for big data applications with the use of tools like Spark – for coding and Hadoop, Cassandra – for data storage.

8 Budget

The completion of the thesis required 600 hours of interning in the company. Additional 100 hours of self-work were devoted for developing a written report. The physical resources used in the thesis comes down to the own computer and company's monitor for desktop extension. The value of these devices usage has been estimated as 10% of their brand new value. Apart from that, all tools were open source software application or services.

TABLE displays the distribution of the working hours and budget allocation.

Table 17 The structure of project's budget.

Activity or Resource	Number of hours dedicated	Value of an hour	Total net value
Development of the application	600	€ 8	€ 4800
Development of the thesis	100	€ 4	€ 400
Laptop	-	-	€ 100
Monitor	-	-	€ 20

The total net budget: € 5320.

The total gross budget, including 21% VAT: € 6437.2

Annex A - Feature Engineering

In the feature engineering phase, there have been chosen two rankings to see which features are the most adequate for clustering problem. The statistical ranking was created based on several metrics. The description of them and graph theory metric is displayed in the table below.

Table 18 Feature Engineering metrics.

Metric	Description	Desirable value for ML program
Coefficient of Variance	Statistical measure of the dispersion of data points in a data series around the mean. The coefficient of variation represents the ratio of the standard deviation to the mean, and it is a useful statistic for comparing the degree of variation from one data series to another.	Higher Better
Variance	Variance is a measurement of the spread between numbers in a data set. The variance measures how far each number in the set is from the mean.	Higher Better
Correlation Coefficient	The correlation coefficient is a measure that determines the degree to which two variables' movements are associated. The range of values for the correlation coefficient is -1.0 to 1.0. A correlation of -1.0 indicates a perfect negative correlation, while a correlation of 1.0 indicates a perfect positive correlation.	Lower Better
Maximal Compression Index (MCI)	MCI encapsulates the information carried by variance and correlation coefficient between two variables. The lower MCI the better given feature is, thereafter Equation 2 has been introduced to unify the scores.	Lower Better
Statistical Method Score	The score is calculated as an average of normalized scores by features in MCI, Variance and Coefficient of Variance.	Higher better
Laplacian Score	Special case of SPEC metrics which instead of RBF kernel function used as a criterion uses other function. LS is very effective and efficient with respect to the data size. Similar to SPEC, the most time consuming in LS is constructing the similarity matrix.	Lower Better

Maximum Compression Index is calculated with the following formula.

Equation 1 Maximum Compression Index equation.

$$2MCI(x,y) = (Var(x) + Var(y) - \sqrt{(Var(x) + Var(y))^2 - 4Var(x) * Var(y) * (1 - Corr(x,y)^2)}$$

Where,

MCI - Maximum Compression Index,

Var – Variance,

Corr – Correlation Coefficient.

Equation 2 MCI Score equation.

$$MCI\ Score = |1 - 2MCI(x, y)\ |$$

The statistical ranking is created with the use of normalized scores of each feature in MCI, Variance and Coefficient of Variance rankings.

Equation 3 Statistical ranking equation.

$$Statistical\ Ranking(feature) = Average(MCI\ Score(feature) + Coeff(feature) + Var(feature))$$

Annex B - Classifiers Optimization

In the optimization of classifiers, the following set of hyperparameters was tested:

1. Support Vector Machine
    a. Kernel: [‘rbf’, ‘poly’, ‘linear’]
    b. Gamma: ['auto', 1e-3, 1e-4]
    c. C: [1, 10, 100]
    d. Probability = [True]
2. Decision Tree Classifier
    a. Criterion: ['gini', 'entropy']
    b. Class Weight: [None, 'balanced']
3. Logistic Regression
    a. Penalty: ['l1', 'l2']
    b. Solver: ['saga', 'liblinear', 'newton-cg', 'sag', 'lbfgs']
    c. C: [1, 10, 100]
    d. Class Weight: [None, 'balanced']
4. K-Neighbours Classifier
    a. Algorithm: ['auto']
    b. Number of neighbours: [5, 15, 50]
5. Linear Discriminant Analysis
    a. Solver: ['svd', 'lsqr', 'eigen']
    b. Shrinkage: [None, 'auto']
6. Gaussian NB – no parameters to be optimized.

Annex C – Results charts

In this Annex, the screenshots of results for clustering, labelling, classification, and pattern recognition are presented hereunder.

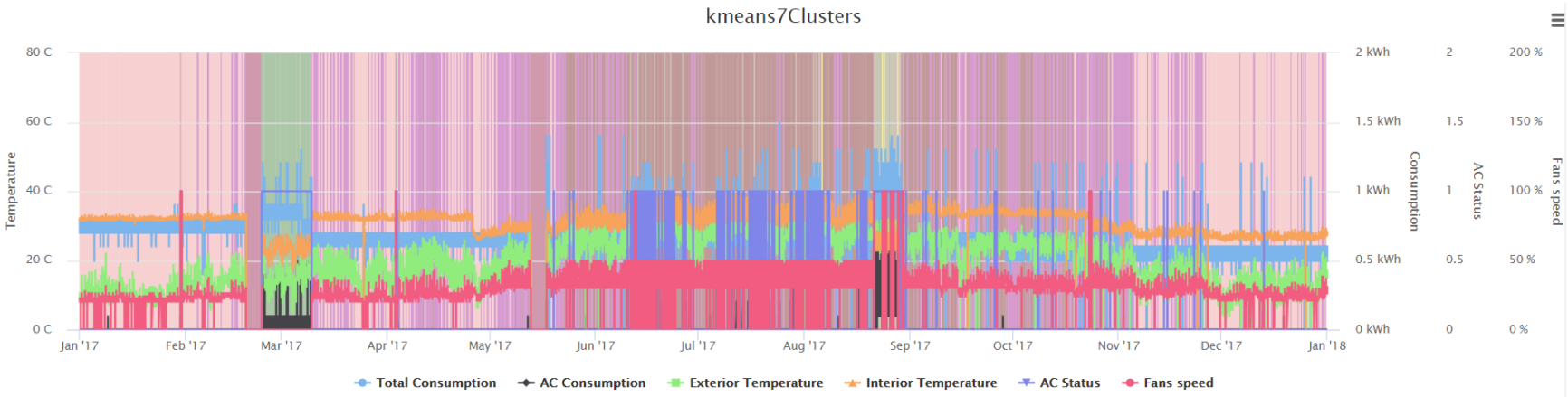

Figure 31 K-Means clustering with 7 clusters chart.

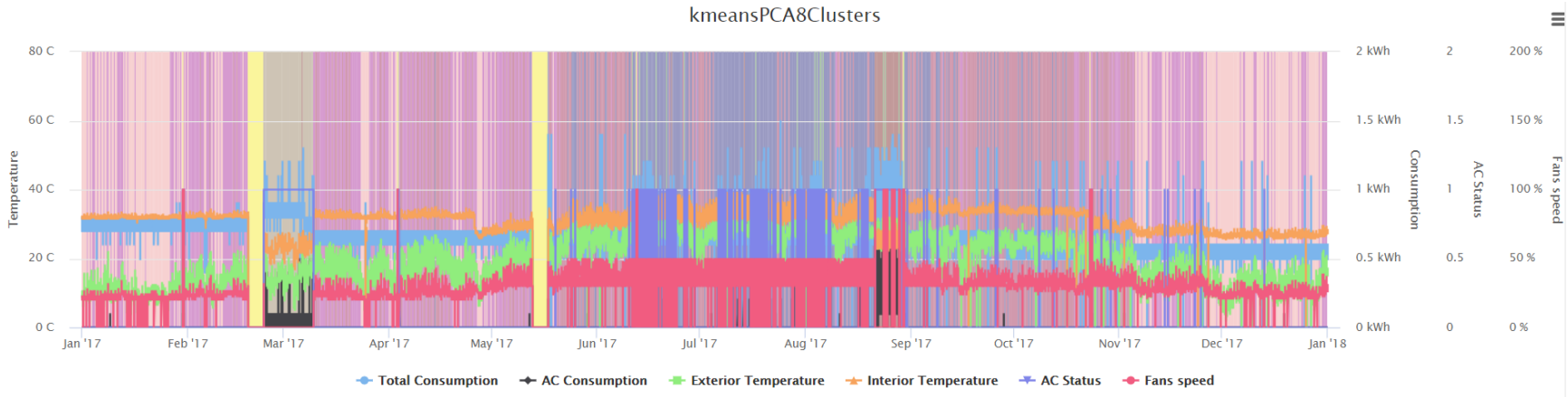

Figure 32 K-Means PCA clustering with 8 clusters chart.

ETSEIB

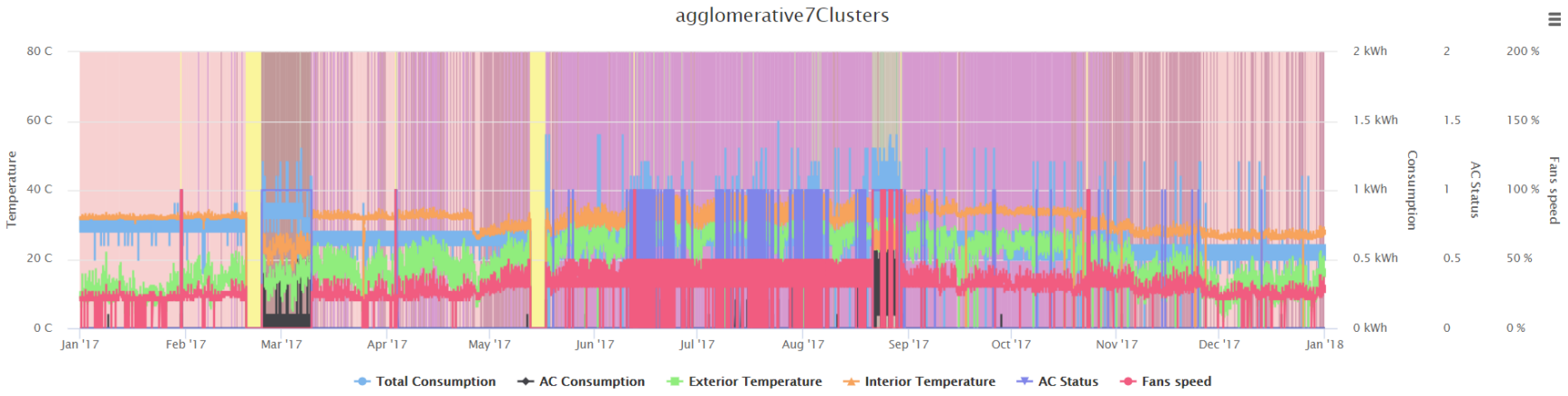

Figure 33 Agglomerative clustering with 7 clusters chart.

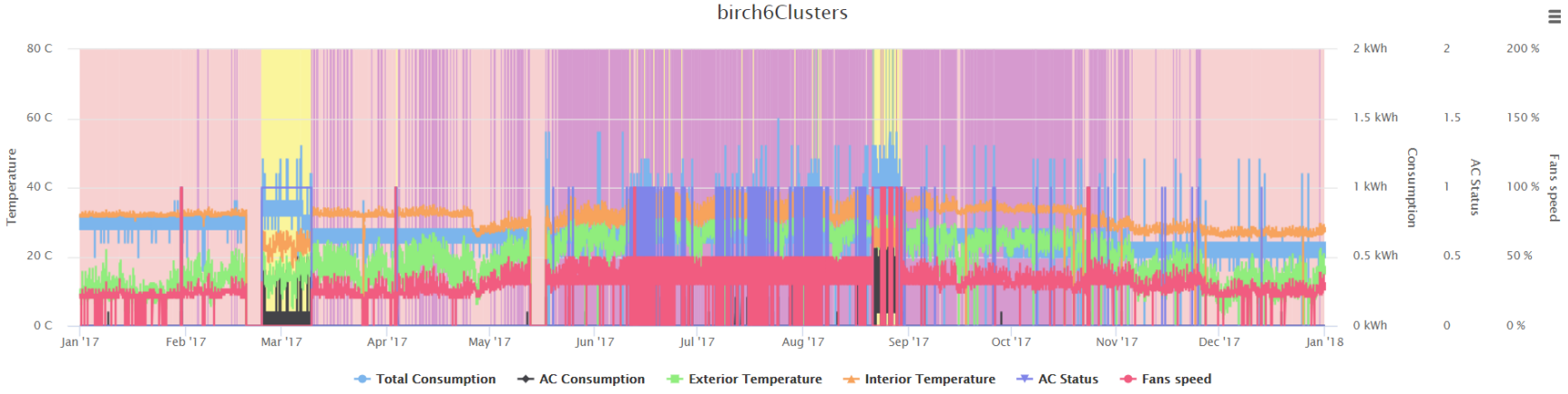

Figure 34 Birch clustering with 6 clusters chart.

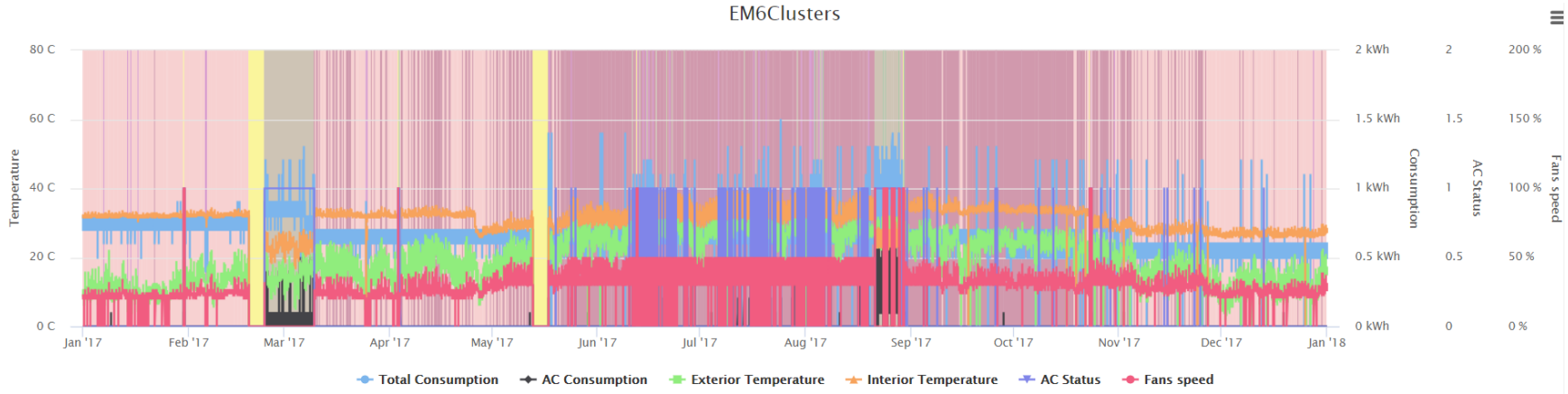

Figure 35 EM clustering with 6 clusters chart.

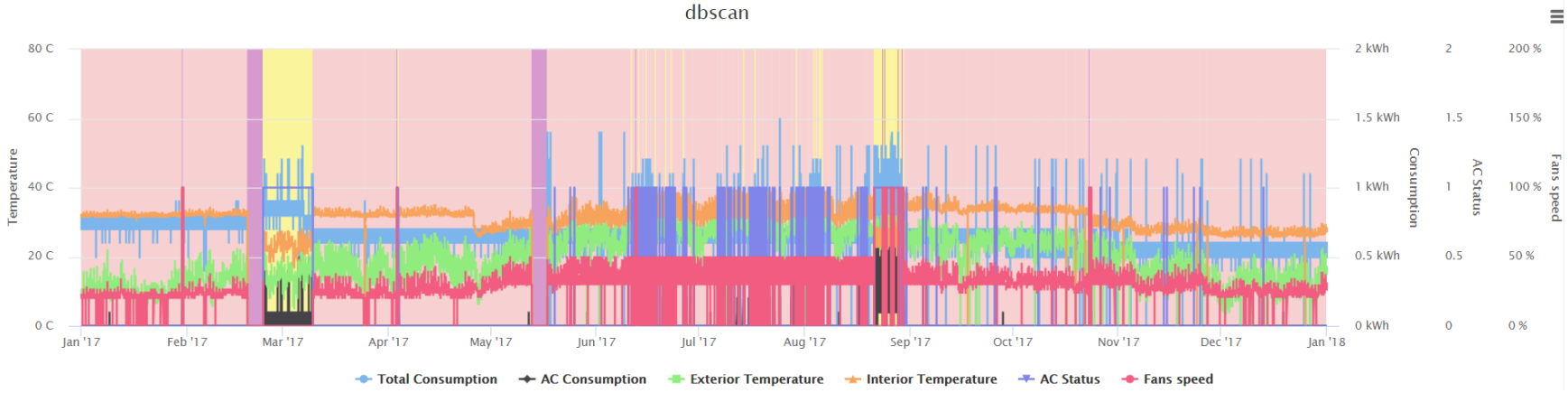

Figure 36 DBSCAN clustering chart.

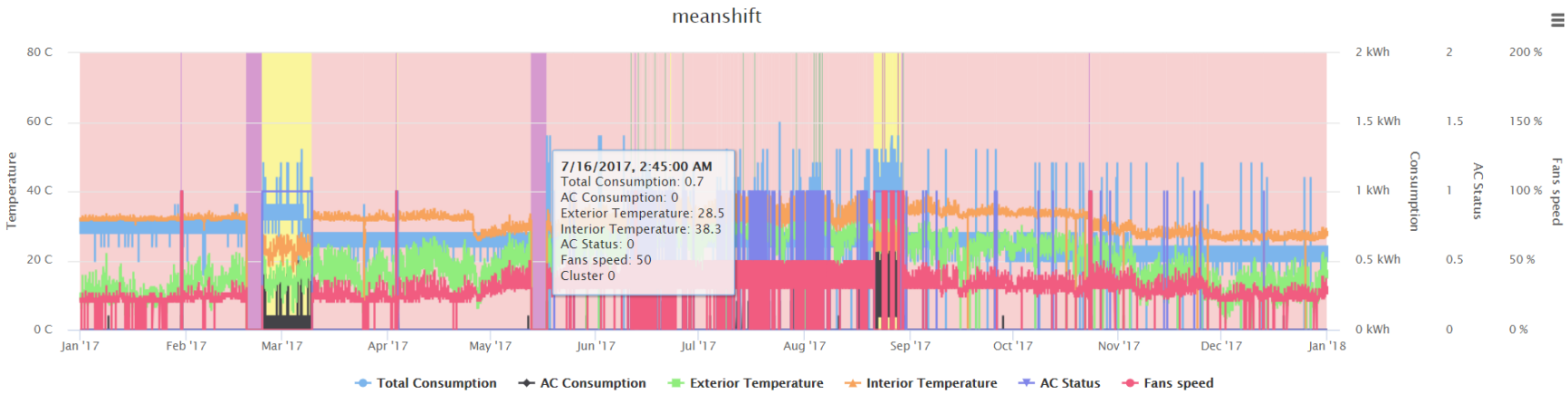

Figure 37 Mean Shift clustering chart with the tooltip displaying the variables and the cluster for given timestamp.

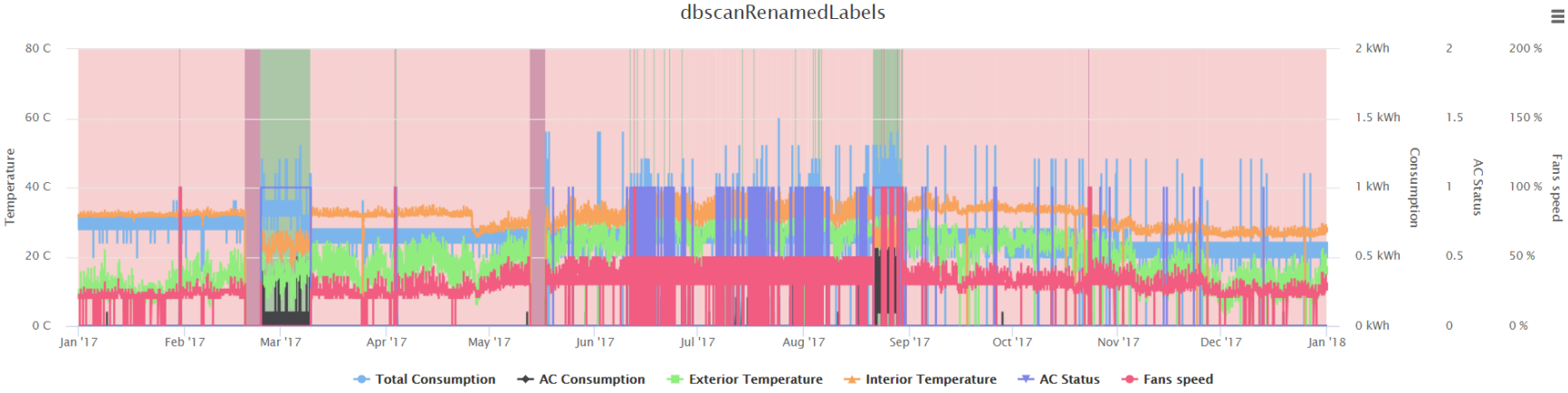

Figure 38 Clustering chart with renamed labels.

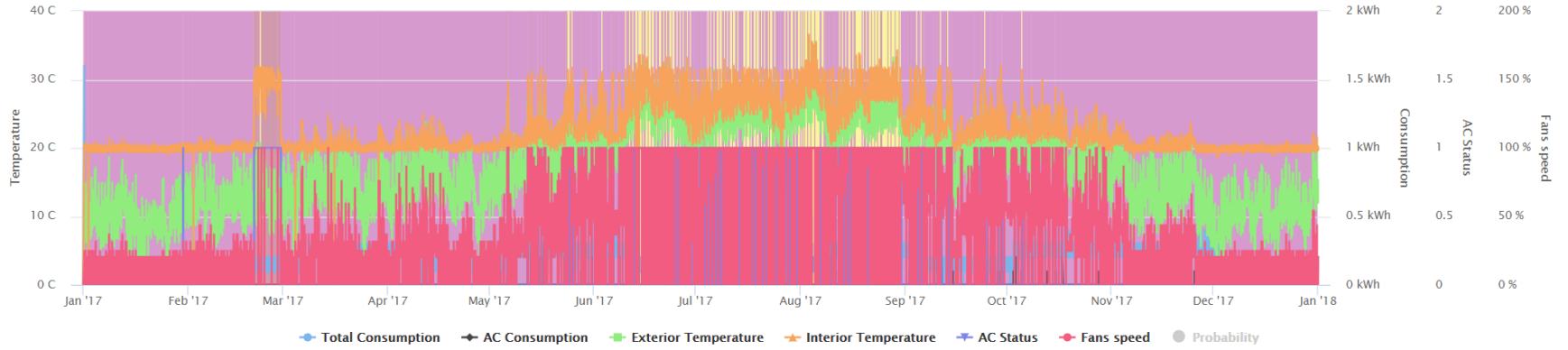

Figure 39 SVM Classification chart of a new BTS.

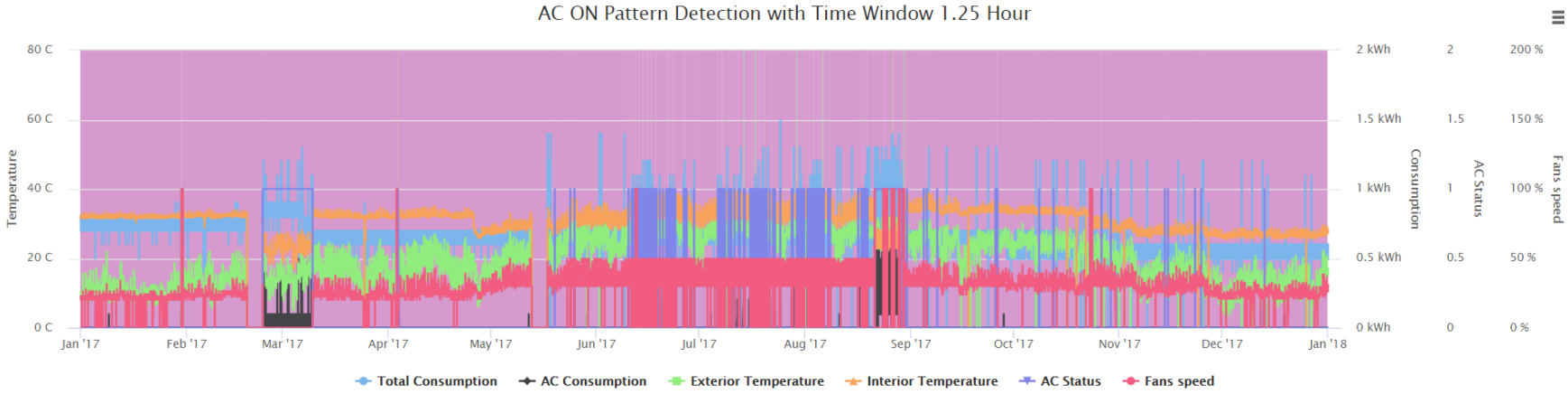

Figure 40 Pattern detection of AC ON mode with 1.25hour time window.

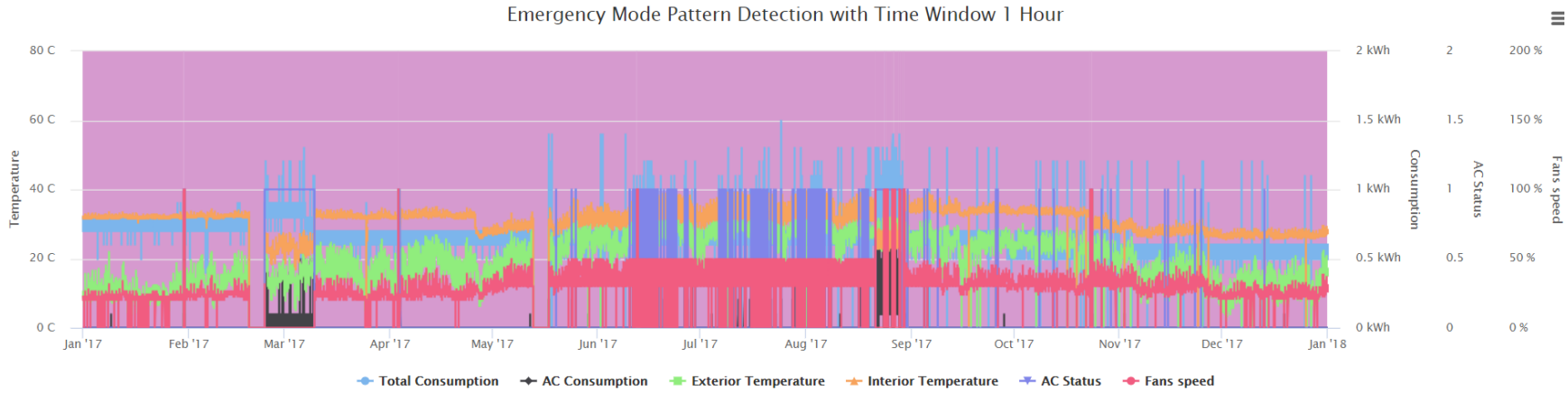

Figure 41 Pattern detection of Emergency Mode with 1 hour time window.

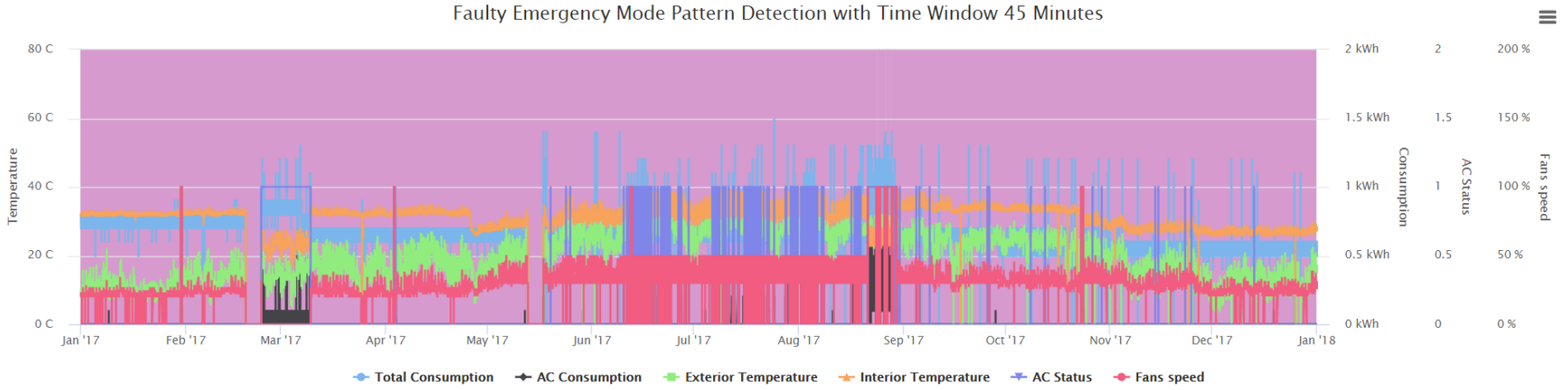

Figure 42 Pattern detection of Faulty Emergency Mode with 45 minutes time window.

References

[1] “IEA,” 2018. [Online]. Available: http://www.iea.org/digital/. [Accessed 14 May 2018].

[2] “BNEF,” January 2018. [Online]. Available: https://about.bnef.com/blog/digitalization-provide-38b-benefits-energy/. [Accessed 14 May 2018].

[3] IEA, “Digitalization & Energy,” 2017.

[4] “LinkedIn,” [Online]. Available: https://www.linkedin.com/company/wattabit/?originalSubdomain=pl. [Accessed 15 May 2018].

[5] S. Arthur, “"Some Studies in Machine Learning Using the Game of Checkers",” *IBM Journal of Research and Development.,* no. 3, p. 210–229, 1959.

[6] M. Swamynathan, Mastering Machine Learning in Six Steps, Bangalore: Apress, 2017.

[7] “UpWork,” [Online]. Available: https://www.upwork.com/hiring/data/sql-vs-nosql-databases-whats-the-difference/. [Accessed 16 May 2018].

[8] “Alooma,” [Online]. Available: https://www.alooma.com/blog/types-of-modern-databases. [Accessed 16 May 2018].

[9] “Envicool,” [Online]. Available: http://www.envicool.com/en/solution/BTScoolingSolution/IntelligentFreeEnergy-saving/default_10_37.aspx. [Accessed 17 May 2018].

[10] “Slideshare,” [Online]. Available: https://www.slideshare.net/naveenjakhar12/gsm-base-transceiver-station. [Accessed 17 May 2018].

[11] “CoolSure,” [Online]. Available: http://www.coolsure.com/index.php/cms/products?name=ecs2000. [Accessed 17 May 2018].

[12] A. P. Antonio Spagnuolo, “MONITORING AND OPTIMIZATION OF ENERGY CONSUMPTION OF BASE TRANSCEIVER STATIONS,” Department of Environmental Science and Technology (DiSTABiF), Second University of Naples, Naples, 2015.

[13] J. T. a. H. L. Salem Alelyani, “Feature Selection for Clustering: A Review”.

[14] “Scikit Learn,” [Online]. Available: http://scikit-learn.org/stable/modules/clustering.html.

[15] “Toward Data Sciene,” [Online]. Available: https://towardsdatascience.com/train-test-split-and-cross-validation-in-python-80b61beca4b6.

[16] “Computer Weekly,” [Online]. Available: https://www.computerweekly.com/news/450418539/Nordic-region-competes-for-datacentre-dominance. [Accessed 28 May 2018].

[17] “IEA - Digitalization Data Centers,” [Online]. Available: http://www.iea.org/digital/#section-5-1. [Accessed 28 May 2018].

[18] “REE - Statistical data for CO2 emissions associated to the power generation,” [Online]. Available: http://www.ree.es/en/statistical-data-of-spanish-electrical-system/statistical-series/national-statistical-series. [Accessed 28 May 2018].

[19] “REE - Electrical markets statistical series,” [Online]. Available: http://www.ree.es/en/statistical-data-of-spanish-electrical-system/statistical-series/national-statistical-series. [Accessed 28 May 2018].

[20] F. P. Ron Kohavi, “"Glossary of terms",” *Machine Learning,* no. 30, p. 271–274, 1998.